2015-05-22

Provably stable, general purpose projection operators for high-order finite difference methods

Kozdon, Jeremy E.

PROVABLY STABLE, GENERAL PURPOSE PROJECTION OPERATORS FOR HIGH-ORDER FINITE DIFFERENCE METHODS

JEREMY E. KOZDON† AND LUCAS C. WILCOX†

Abstract. A methodology for handling block-to-block coupling of non-conforming, multiblock summation-by-parts finite difference methods is proposed. The coupling is based on the construction of projection operators that move a finite difference grid solution along an interface to a space of piecewise defined function; we specifically consider piecewise polynomial functions. The constructed projection operators are consistent with the underlying summation-by-parts energy norm. Using the linear wave equation in two dimensions as a model problem, energy stability of the coupled numerical method is proven for the case of curved, non-conforming block-to-block interfaces. To further demonstrate the power of the coupling procedure, we show how it allows for the developments of a provably energy stable coupling between curvilinear finite difference methods and a curved-triangle discontinuous Galerkin method. The theoretical results are verified through numerical simulations on curved meshes as well as eigenvalue analysis.

Key words. summation-by-parts, weak enforcement, high-order finite difference methods, coupling, stability, accuracy, projection operator, variational form, interface

AMS subject classifications. 65M06, 65M12, 65M50, 65M60, 65M70

1. Introduction.

Even though high-order multiblock finite difference methods are well suited for many problems, limitations arise for particularly complex geometries. For instance, most formulations require that grids conform at multiblock interfaces, that is the grids lines must be continuous. This poses a challenge since resolution constraints in one portion of the domain can result in unnecessarily high resolution elsewhere in the domain. Furthermore, even though coordinate transforms enable the use of high-order finite difference methods for complex geometries, well-conditioned partitioning of complex domains into quadrilaterals in two-dimensions and hexahedra in three-dimensions can be challenging and/or impossible. The impact of this is a poorly conditioned Jacobian which has an adverse impact on the time step size and truncation error.

One approach to overcome these complications is to relax the requirement that the grid and numerical methods conform across block interfaces. To do this a variety of interpolation and projection techniques have been proposed including the use of overlapping grids [2], strong enforcement using ghost points [18], or weak-enforcement of continuity at block boundaries [11]. Here we particularly highlight the approach of Mattsson and Carpenter [11] as it is closely related to the work presented below. In that paper, stability conditions were presented for interpolation operators between summation-by-parts (SBP) finite difference methods and operators were constructed for several fixed refinement ratio interfaces. The operators proposed by Mattsson and Carpenter required that at the block level the interfaces conformed (i.e., the corners had to match), a restriction which has been removed by the work of Nissen, Kormann, Grandin, and Virta [13] (though this later work retained the fixed refinement ratio requirement). It has also been proposed to couple high-order finite difference methods with unstructured grid methods as another approach to avoiding geometric constraints. For example, Nordström and Gong [16] proposed coupling a high-order SBP method with an unstructured second-order finite volume method.

†Department of Applied Mathematics, Naval Postgraduate School, Monterey, CA, 93943–5216 ({jekozdon,lwilcox}@nps.edu)

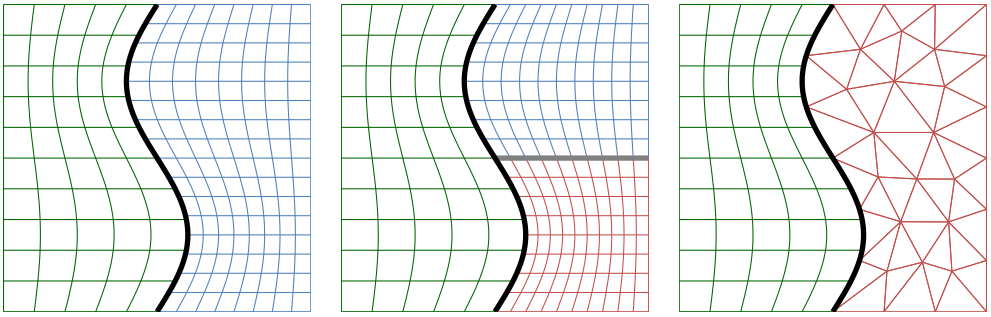

FIG. 1.1. *Illustration of the computational grids supported by the projection operators described in this paper. On the left is a conforming two-block SBP grid. In the middle is a nonconforming three-block SBP grid; here nonconforming refers to the SBP blocks and not the SBP grids. On the right is a coupled structured-unstructured grid; in this paper the methods used will be SBP finite differences and DG grid. In each illustration the thick line between blocks indicates coupling interfaces, i.e., the locations of the glue grids.*

The class of finite difference methods to be considered in this work are SBP finite difference methods [8, 9, 19, 12]; see § 2 for the basic SBP ideas used in this work. One of the important features of SBP methods is that the difference operators have an associated energy norm which discretely mimics integration by parts; this is referred to as the SBP property.

Here we present a general purpose technique for handling block-to-block coupling of non-conforming multiblock SBP finite difference methods. This technique also allows for the coupling with unstructured methods such as the discontinuous Galerkin (DG) method. The types of grid couplings possible are illustrated in Fig. 1.1. This coupling procedure uses projection operators that move finite difference grid solutions along the coupling interface to piecewise polynomial functions. It is with respect to the built-in norm of the SBP finite difference methods that the projection operators are constructed. The SBP property alone is not sufficient to guarantee stability, SBP preserving boundary and interface closures are also required. In this work we will achieve this by enforcing all boundary and interface conditions weakly through the so-called simultaneous approximation term (SAT) method [1]; this is similar to the use of numerical flux terms in the DG method, a fact which will be exploited to stably couple SBP and DG methods.

Since the projection operators move the solution to a piecewise continuous representation (where projections are straight forward to construct), the operators only need to be constructed once for each finite difference operator. That is projection operators can be constructed independent of the numerical method and grid on the other side of an interface. This independence of interface type is one of the features that enables both development of provably stable couplings between conforming and nonconforming SBP meshes as well as between SBP and DG methods. The fact that nonconforming grids can be accommodated enables the development of adaptive mesh refinement codes using high-order, summation-by-parts finite difference methods.

For simplicity of presentation, we take as our model problem the two-dimensional linear acoustic wave equation in first order form. We prove that the proposed coupling is stable for this system of equation as well as provide numerical evidence to confirm the analytical result. Since the projection operators are constructed based on the SBP operator, and do not depend on the system of equations being solved, using

standard techniques the extension to other linear symmetric hyperbolic systems should be possible.

2. Definitions. We begin by stating a few preliminary definitions that are at the heart of this work.

DEFINITION 2.1 (SBP property). *A difference approximation* $\boldsymbol{D}$ *is called a summation-by-parts (SBP) approximation to* d/dx *if it can be decomposed as* $\boldsymbol{D} = \boldsymbol{H}^{-1}\boldsymbol{Q}$ *with* $\boldsymbol{H}$ *being positive definite and* $\boldsymbol{Q}$ *having the property* $\boldsymbol{Q} + \boldsymbol{Q}^T = \boldsymbol{B} = \text{diag}\begin{bmatrix}-1 & 0 & \cdots & 0 & 1\end{bmatrix}$*, i.e.,* $\boldsymbol{Q}$ *is almost skew-symmetric.*

To understand why such a difference approximation is called SBP consider a grid function $\boldsymbol{f} = \begin{bmatrix}f_0 & f_1 & \cdots & f_N\end{bmatrix}^T$. The $\boldsymbol{H}$-weighted inner product of $\boldsymbol{f}$ and $\boldsymbol{Df}$ gives

$$(\boldsymbol{f}, \boldsymbol{Df})_H = \boldsymbol{f}^T\boldsymbol{HDf} = \boldsymbol{f}^T\boldsymbol{Qf} = \frac{1}{2}\boldsymbol{f}^T\left(\boldsymbol{Q} + \boldsymbol{Q}^T\right)\boldsymbol{f} = \frac{1}{2}\left(f_N^2 - f_0^2\right), \tag{2.1}$$

which is of the same form as the inner product of a continuously differentiable function $f \in C^1[x_l, x_r]$ and df/dx:

$$\left(f, \frac{df}{dx}\right)_{L^2(x_l,x_r)} = \int_{x_l}^{x_r} f\frac{df}{dx}\,dx = \frac{1}{2}\left(f_r^2 - f_l^2\right), \tag{2.2}$$

where (x_l, x_r) is an open interval of the real line $\mathbb{R}$, $f_l = f(x_l)$, and $f_r = f(x_r)$.

The difference operators commonly referred to as SBP methods are central difference operators in the interior (with orders: 2, 4, 6, 8, ...) which transition to one-sided approximation near the boundary in such a way that the SBP property is achieved [8, 9, 12, 19]. This transition to one-sided typically leads to a degradation in accuracy at the boundary. These SBP operators are subdivided into two classes: diagonal norm (diagonal $\boldsymbol{H}$) and block norm (non-diagonal $\boldsymbol{H}$) operators. For the diagonal norm operators the boundary accuracy can be at most half the interior accuracy, i.e., if the approximation is $2q$-accurate in the interior it is at most q-accurate at the boundary. For the block norm operators it is possible to construct difference approximations that are $2q-1$ accurate at the boundary. In both cases the global accuracy of the scheme is one more than the boundary accuracy, i.e., $q+1$ for the diagonal norm operators and $2q$ for the block norm operators [3]. For most practical calculations the diagonal norm operators are used as they result in stable scheme for both coordinate transforms and problems with variable coefficients [6, 7, 14, 15, 17]; a notable exception is the recent work of Mattsson and Almquist [10] where artificial dissipation is used to stabilize the block norm operators in complex geometries.

A key concept for this work is the definition of an *SBP* $\boldsymbol{H}$*-compatible projection operator*. This operator will allow us to move from a grid function to a space of piecewise continuous functions in a manner that is compatible (in an $L^2(\Gamma)$ sense) with the SBP finite difference method. We call the space of piecewise continuous functions the *glue grid* since it allows us to "glue" together differing numerical methods.

To make this more concrete, given a finite difference grid $\begin{bmatrix}x_0 & x_1 & \cdots & x_N\end{bmatrix}$ let $\mathcal{G}_h \subset L^2(\Gamma)$ be a finite-dimensional space of functions, i.e., the space of functions the glue grid can represent. Let $\boldsymbol{\varphi}(\eta) = \begin{bmatrix}\varphi_0(\eta) & \varphi_1(\eta) & \cdots & \varphi_K(\eta)\end{bmatrix}^T$ be a vector of linearly independent basis functions for $\mathcal{G}_h$, $\boldsymbol{f} = \begin{bmatrix}f_0 & f_1 & \cdots & f_N\end{bmatrix}^T$ be a grid function, and $\boldsymbol{H}$ be an SBP norm. Our goal is to define a projection operator such that a set of coefficients $\bar{\boldsymbol{f}} = \begin{bmatrix}\bar{f}_0 & \bar{f}_1 & \cdots & \bar{f}_K\end{bmatrix}^T$ can be defined from $\boldsymbol{f}$ such that $\bar{f}(\eta) = \sum_i \bar{f}_i\varphi_i(\eta) = \bar{\boldsymbol{f}}^T\boldsymbol{\varphi}(\eta)$ is a compatible representation of the grid function $\boldsymbol{f}$

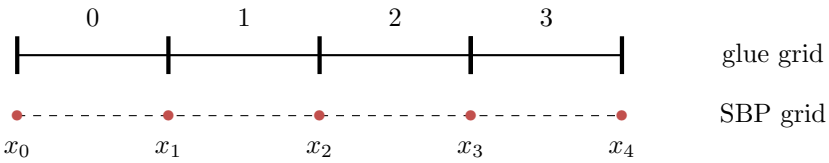

FIG. 2.1. *Alignment of the glue grid, the line with interval boundaries denoted with hatch marks, and the finite difference grid, denoted with dots representing the grid points, at the leftmost boundary. The glue grid supports functions which are continuous on each interval, e.g., piecewise continuous functions. Note that the glue grid and the finite difference grid coincide spatially but here we have separated them vertically for display purposes.*

in the space $\mathcal{G}_h$. Note that throughout the paper we use the overline notation to represent quantities defined on the glue grid.

In order to define the $\boldsymbol{H}$-compatible projection operators we must define the mass matrix on the glue grid. Namely, the symmetric, positive definite mass matrix is $\boldsymbol{M} = \int_\Gamma \boldsymbol{\varphi}(\eta)\boldsymbol{\varphi}^T(\eta)\, d\eta$; thus given two functions $\bar{f}(\eta) = \bar{\boldsymbol{f}}^T\boldsymbol{\varphi}(\eta)$ and $\bar{g}(\eta) = \bar{\boldsymbol{g}}^T\boldsymbol{\varphi}(\eta)$ in $\mathcal{G}_h$ the inner product is $(\bar{f}, \bar{g}) = \bar{\boldsymbol{f}}^T\boldsymbol{M}\bar{\boldsymbol{g}}$.

DEFINITION 2.2 ($\boldsymbol{H}$-Compatible Projection Operator). *Let $\boldsymbol{f}$ be a grid function and $\bar{u}(\eta) = \bar{\boldsymbol{u}}^T\boldsymbol{\varphi}(\eta) \in \mathcal{G}_h$ be a glue grid function. We call the projection matrices $\boldsymbol{P}_{f2g}$ and $\boldsymbol{P}_{g2f}$ $\boldsymbol{H}$-compatible if for all $\boldsymbol{f}$ and $\bar{\boldsymbol{u}}$:*

$$\boldsymbol{u}^T\boldsymbol{H}\boldsymbol{f} = \bar{\boldsymbol{u}}^T\boldsymbol{M}\bar{\boldsymbol{f}}, \tag{2.3}$$

where $\bar{\boldsymbol{f}} = \boldsymbol{P}_{f2g}\boldsymbol{f}$ and $\boldsymbol{u} = \boldsymbol{P}_{g2f}\bar{\boldsymbol{u}}$, or equivalently

$$\boldsymbol{P}_{g2f}^T\boldsymbol{H} = \boldsymbol{M}\boldsymbol{P}_{f2g}. \tag{2.4}$$

Here the subscript $f2g$ stands for projection from the finite difference grid to the glue grid and $g2f$ from the glue grid to the finite difference grid.

Notice, that nothing in the definition implies that these solutions must be accurate representations of one another and Definition 2.2 will only be used to guarantee stability. Furthermore, there is no statement that the functions can be moved between spaces without error, that is we do not assume that $\boldsymbol{P}_{g2f}\boldsymbol{P}_{f2g}\boldsymbol{v} = \boldsymbol{v}$ nor that $\boldsymbol{P}_{f2g}\boldsymbol{P}_{g2f}\bar{\boldsymbol{u}} = \bar{\boldsymbol{u}}$.

It is natural to augment Definition 2.2 with a set of accuracy conditions based on the particular glue grid space $\mathcal{G}_h$ being used. In this work, we let $\mathcal{G}_h$ be the space of piecewise polynomials where we let the intervals over which the polynomial are defined align with the finite difference points as shown in Fig. 2.1. Motivated by Mattsson and Carpenter [11], we require that the operators used in this work satisfy a set of polynomial accuracy conditions. Namely with a glue grid that can represent qth order polynomials exactly, we define the qth order polynomial grid function as $\boldsymbol{f}_q = \begin{bmatrix} 0^q & 1^q & \cdots & N^q \end{bmatrix}^T$ (with the convention that $0^0 = 1$) and let $\bar{g}_q(\eta) = \bar{\boldsymbol{g}}_q^T\boldsymbol{\varphi}(\eta)$ be the same polynomial on the glue grid. We then require that the errors

$$\boldsymbol{e}_{g2f} = \boldsymbol{P}_{g2f}\bar{\boldsymbol{g}}_q - \boldsymbol{f}_q, \qquad \boldsymbol{e}_{f2g} = \boldsymbol{P}_{f2g}\boldsymbol{f}_q - \bar{\boldsymbol{g}}_q \tag{2.5}$$

be zero for all polynomials up to order $q_i - 1$ everywhere except near the boundary where it is required the error be zero for polynomials up to order $q_b - 1$; here q_i and q_b are the interior and boundary accuracy of the SBP finite difference method being

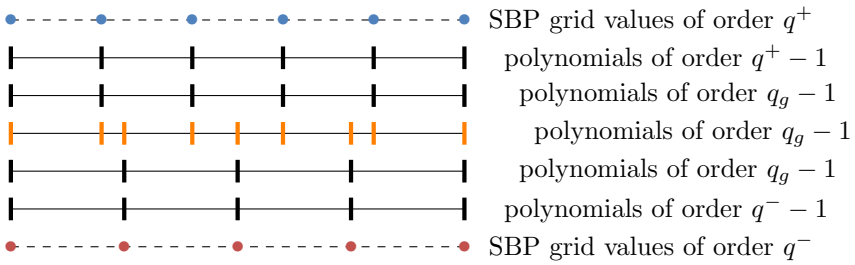

FIG. 2.2. *Example of the alignment of a glue grid with two SBP finite difference grids. On the finite difference grids values are stored at the nodal locations whereas the glue grids stores polynomials over the indicated intervals. The projection operators associated with the SBP finite difference grid are defined to go to glue grid that conform to the SBP grid nodes. Since both sides of the interface do not have the same space of piecewise polynomials, additional projection operators are needed to move between the polynomial spaces.*

used. In other words, we require the projection operators to mimic the accuracy of the SBP finite difference method. These are the same accuracy conditions used in the finite difference to finite difference operators of Mattsson and Carpenter [11]. The construction of projection operators that satisfy the above accuracy and stability conditions using a space of piecewise polynomials are discussed in Appendix B.

We emphasize again that the only requirement for a projection operator to result in a stable discretization is Definition 2.2. The accuracy conditions (2.5) only pertain to the specific choice of $\mathcal{G}_h$ in this paper and other conditions may be required for different glue grid spaces.

The $\boldsymbol{H}$-compatible projection operators discussed in Appendix B move between an SBP finite difference solution and a given set of piecewise polynomials of order $q-1$ where q is the accuracy of the finite difference method; see for example Fig. 2.1. To make these operators useful in practice we need to be able to transition between glue grid spaces.

To understand why this is, consider the situation shown in Fig. 2.2. Here an interface between two SBP operators with different grids and orders of accuracy is shown. Since the operators constructed in Appendix B move between finite difference grid values and a fixed set of piecewise continuous polynomials (fixed intervals and orders), the glue grids defined for either side of the interface in Fig. 2.2 will not conform (i.e., the polynomials may be of a different order and/or the locations of the interval boundaries may be different). Thus, additional projection operators are needed to move between the different polynomial orders and intervals.

The projection operators in this work are constructed in a hierarchical fashion where at each stage we need to construct projections between two different piecewise polynomial glue grid spaces where one is the subset of the other, i.e., $G_h^a \subset G_h^b$. The projection operators between spaces are constructed to satisfy an analog of (2.4), namely

$$\boldsymbol{M}_a \boldsymbol{P}_{g_b 2 g_a} = \boldsymbol{P}_{g_a 2 g_b}^T \boldsymbol{M}_b, \tag{2.6}$$

where $\boldsymbol{M}_a$ and $\boldsymbol{M}_b$ are the respective glue grid mass matrices (which are each symmetric positive definite given linearly independent basis functions). Since $G_h^a \subset G_h^b$ then $\boldsymbol{P}_{g_a 2 g_b}$ is a basis transformation operation which can be determined independent

of $\boldsymbol{P}_{g_b 2 g_a}$ and we have

$$\boldsymbol{P}_{g_b 2 g_a} = \boldsymbol{M}_a^{-1} \boldsymbol{P}_{g_a 2 g_b}^T \boldsymbol{M}_b. \tag{2.7}$$

LEMMA 2.3. *Suppose $\boldsymbol{P}_{f2g_a}$ and $\boldsymbol{P}_{g_a 2 f}$ satisfy (2.4) with $\boldsymbol{M} = \boldsymbol{M}_a$ and $\boldsymbol{P}_{g_b 2 g_a}$ and $\boldsymbol{P}_{g_a 2 g_b}$ satisfy (2.7), then*

$$\boldsymbol{P}_{f2g_b} = \boldsymbol{P}_{g_a 2 g_b} \boldsymbol{P}_{f2g_a}, \qquad \boldsymbol{P}_{g_b 2 f} = \boldsymbol{P}_{g_a 2 f} \boldsymbol{P}_{g_b 2 g_a} \tag{2.8}$$

satisfy (2.4) with $\boldsymbol{M} = \boldsymbol{M}_b$.

Proof. By direct calculation we have

$$\begin{aligned} \boldsymbol{P}_{g_b 2 f}^T \boldsymbol{H} = \boldsymbol{P}_{g_b 2 g_a}^T \boldsymbol{P}_{g_a 2 f}^T \boldsymbol{H} &= \boldsymbol{P}_{g_b 2 g_a}^T \boldsymbol{M}_a \boldsymbol{P}_{f2g_a} \\ &= \boldsymbol{M}_b \boldsymbol{P}_{g_a 2 g_b} \boldsymbol{P}_{f2g_a} = \boldsymbol{M}_b \boldsymbol{P}_{f2g_b} \end{aligned} \tag{2.9}$$

□

Further, it follows from the nesting of the glue spaces that the accuracy conditions (2.5) are also satisfied by these composition of projection operators. Due this and Lemma 2.3 we may assume without loss of generality that $\boldsymbol{P}_{f2g}$ and $\boldsymbol{P}_{g2f}$ project all the way through to the finest glue space (represented by the middle glue grid in Fig. 2.2), that is the intermediate spaces are not explicitly considered further in this work.

3. Acoustic Wave Equation: SBP-SAT Discretization. As a model problem we consider the two-dimensional acoustic wave equation in first order form:

$$\rho \frac{\partial v_i}{\partial t} + \frac{\partial p}{\partial x_i} = 0 \ (i = 1, 2), \qquad \frac{\partial p}{\partial t} + \lambda \left(\frac{\partial v_1}{\partial x_1} + \frac{\partial v_2}{\partial x_2} \right) = 0, \tag{3.1}$$

where v_1 and v_2 are the particle velocities in the x_1 and x_2 directions, respectively, and p is the pressure. Here, ρ is the material density and λ is Lamé's second parameter where we assume $\rho, \lambda > 0$.

We are interested in discretizing (3.1) on a domain Ω which is the union of curvilinear, quadrilateral domains (blocks) $\{\Omega_e\}$. To do so, we transform each domain from physical space Ω_e to the reference space $\tilde{\Omega} = [-1, 1] \times [-1, 1]$ via the coordinate transform $x_i = x_i(\xi_1, \xi_2)$, $i = 1, 2$, with x_i being the coordinates in physical domain and ξ_i being the coordinates in the reference domain; we assume that the inverse transforms $\xi_i(x_1, x_2)$ also exist. (Note, for simplification of notation we suppress the geometrical terms dependence on each domain.) The Jacobian determinant is

$$J = \frac{\partial x_1}{\partial \xi_1} \frac{\partial x_2}{\partial \xi_2} - \frac{\partial x_2}{\partial \xi_1} \frac{\partial x_1}{\partial \xi_2}, \tag{3.2}$$

which gives rise to the metric relations

$$J \frac{\partial \xi_1}{\partial x_1} = \frac{\partial x_2}{\partial \xi_2}, \qquad J \frac{\partial \xi_1}{\partial x_2} = -\frac{\partial x_1}{\partial \xi_2}, \qquad J \frac{\partial \xi_2}{\partial x_2} = \frac{\partial x_1}{\partial \xi_1}, \qquad J \frac{\partial \xi_2}{\partial x_1} = -\frac{\partial x_2}{\partial \xi_1}. \tag{3.3}$$

With these definitions, the acoustic wave equation (3.1) can be written as

$$\rho J \frac{\partial v_i}{\partial t} + \frac{\partial}{\partial \xi_1} \left(J \frac{\partial \xi_1}{\partial x_i} p \right) + \frac{\partial}{\partial \xi_2} \left(J \frac{\partial \xi_2}{\partial x_i} p \right) = 0, \quad i = 1, 2, \tag{3.4}$$

$$J \frac{\partial p}{\partial t} + \lambda \left(J \frac{\partial \xi_1}{\partial x_1} \frac{\partial v_1}{\partial \xi_1} + J \frac{\partial \xi_2}{\partial x_1} \frac{\partial v_1}{\partial \xi_2} + J \frac{\partial \xi_1}{\partial x_2} \frac{\partial v_2}{\partial \xi_1} + J \frac{\partial \xi_2}{\partial x_2} \frac{\partial v_2}{\partial \xi_2} \right) = 0. \tag{3.5}$$

Notice that we have written the transformed equations in skew-symmetric form with the velocity equations (3.4) written using a conservative transform and the pressure equation (3.5) using a non-conservative transform. It is common to do both terms conservatively, but doing this splitting results in a provably stable scheme; a similar splitting was introduced by Kozdon, Dunham, and Nordström [7] as a method for achieving provably stability.

Before presenting an SBP discretization of the governing equations, we first introduce the variational form of the equations on each domain. This is done to highlight the close connection between SBP finite difference methods and DG methods. To do this we introduce test functions w_i, $i = 1, 2$, and φ which belong to some appropriately chosen space. Multiplying the velocity equation (3.4) by w_i, the pressure equation (3.5) by φ, and integrating over a domain $\tilde{\Omega}$ gives

$$\begin{aligned}
&\int_{\tilde{\Omega}} w_i \left[\rho J \frac{\partial v_i}{\partial t} + \frac{\partial}{\partial \xi_1}\left(J\frac{\partial \xi_1}{\partial x_i}p\right) + \frac{\partial}{\partial \xi_2}\left(J\frac{\partial \xi_2}{\partial x_i}p\right)\right] dA \\
&\qquad = -\int_{\partial\tilde{\Omega}} w_i S_J n_i \left(p^* - p\right) ds, \quad i = 1, 2, \qquad (3.6)\\
&\int_{\tilde{\Omega}} \varphi \left[J\frac{\partial p}{\partial t} + \lambda\left(J\frac{\partial \xi_1}{\partial x_1}\frac{\partial v_1}{\partial \xi_1} + J\frac{\partial \xi_2}{\partial x_1}\frac{\partial v_1}{\partial \xi_2} + J\frac{\partial \xi_1}{\partial x_2}\frac{\partial v_2}{\partial \xi_1} + J\frac{\partial \xi_2}{\partial x_2}\frac{\partial v_2}{\partial \xi_2}\right)\right] dA \\
&\qquad = -\int_{\partial\tilde{\Omega}} \varphi\lambda S_J \left(v^* - v\right) ds. \qquad (3.7)
\end{aligned}$$

where p^* and v^* are newly introduced states satisfying the boundary or interface conditions that connect the domains; these are related to the numerical fluxes defined below. Here, S_J is the surface Jacobian, n_1 and n_2 are the components of the outward pointing unit normal (in the x_1 and x_2 directions, respectively), and $v = n_1 v_1 + n_2 v_2$ normal component of velocity. For the reference domain $\tilde{\Omega} = [-1, 1] \times [-1, 1]$ (which will be used for the finite difference discretization) the surface Jacobian terms and outward pointing normals for the edge defined by $\xi_i = \pm 1$ are

$$S_J = J\sqrt{\left(\frac{\partial \xi_i}{\partial x_2}\right)^2 + \left(\frac{\partial \xi_i}{\partial x_1}\right)^2}, \quad n_1 = \pm\frac{J}{S_J}\frac{\partial \xi_i}{\partial x_2}, \quad n_2 = \pm\frac{J}{S_J}\frac{\partial \xi_i}{\partial x_1}. \qquad (3.8)$$

With the above definition, the boundary integrals can then be rewritten as

$$\begin{aligned}
&\int_{\partial\tilde{\Omega}} w_i S_J n_i \left(p^* - p\right) ds \\
&= \int_{-1}^{1} \left[w_i S_J n_i \left(p^* - p\right)\right]_{\xi_1=-1} d\xi_2 + \int_{-1}^{1} \left[w_i S_J n_i \left(p^* - p\right)\right]_{\xi_1=1} d\xi_2 \qquad (3.9)\\
&\quad + \int_{-1}^{1} \left[w_i S_J n_i \left(p^* - p\right)\right]_{\xi_2=-1} d\xi_1 + \int_{-1}^{1} \left[w_i S_J n_i \left(p^* - p\right)\right]_{\xi_2=1} d\xi_1,
\end{aligned}$$

$$\begin{aligned}
&\int_{\partial\tilde{\Omega}} \varphi\lambda S_J \left(v^* - v\right) ds \\
&= \int_{-1}^{1} \left[\varphi\lambda S_J \left(v^* - v\right)\right]_{\xi_1=-1} d\xi_2 + \int_{-1}^{1} \left[\varphi\lambda S_J \left(v^* - v\right)\right]_{\xi_1=1} d\xi_2 \qquad (3.10)\\
&\quad + \int_{-1}^{1} \left[\varphi\lambda S_J \left(v^* - v\right)\right]_{\xi_2=-1} d\xi_1 + \int_{-1}^{1} \left[\varphi\lambda S_J \left(v^* - v\right)\right]_{\xi_2=1} d\xi_1.
\end{aligned}$$

Going back to the differential form of the equations, we discretize the reference domain $\tilde{\Omega}$ with an $(N_1+1)\times(N_2+1)$ grid of equally spaced points. The grid spacing in the ξ_i dimension is $h_i = 2/N_i$ for $i = 1, 2$. Thus the (k, l) grid point is at $(\xi_1, \xi_2) = (kh_1 - 1, lh_2 - 1)$ for $k = 0, \ldots, N_1$ and $l = 0, \ldots, N_2$. We define the pressure solution vector on the grid as

$$\boldsymbol{p} = \begin{bmatrix} p_{00} & p_{01} & \cdots & p_{0N_2} & p_{10} & \cdots & p_{N_1N_2} \end{bmatrix}^T, \tag{3.11}$$

where p_{kl} approximates the pressure p at grid point (k, l); the solution vectors $\boldsymbol{v}_1$ and $\boldsymbol{v}_2$ are similarly defined. An SBP-SAT semi-discretization (discretization only in space) of (3.4)–(3.5) on a domain Ω_e using an $(N_1+1)\times(N_2+1)$ grid is then

$$\rho \boldsymbol{J}\frac{d\boldsymbol{v}_i}{dt} + \boldsymbol{D}_1\boldsymbol{J}\frac{\partial \boldsymbol{\xi}_1}{\partial \boldsymbol{x}_i}\boldsymbol{p} + \boldsymbol{D}_2\boldsymbol{J}\frac{\partial \boldsymbol{\xi}_2}{\partial \boldsymbol{x}_i}\boldsymbol{p} = -\boldsymbol{H}^{-1}\boldsymbol{\mathcal{F}}_{v_i}, \quad i = 1, 2, \tag{3.12}$$

$$\boldsymbol{J}\frac{\partial \boldsymbol{p}}{\partial t} + \lambda\left(\boldsymbol{J}\frac{\partial \boldsymbol{\xi}_1}{\partial \boldsymbol{x}_1}\boldsymbol{D}_1\boldsymbol{v}_1 + \boldsymbol{J}\frac{\partial \boldsymbol{\xi}_2}{\partial \boldsymbol{x}_1}\boldsymbol{D}_2\boldsymbol{v}_1 + \boldsymbol{J}\frac{\partial \boldsymbol{\xi}_1}{\partial \boldsymbol{x}_2}\boldsymbol{D}_1\boldsymbol{v}_2 + \boldsymbol{J}\frac{\partial \boldsymbol{\xi}_2}{\partial \boldsymbol{x}_2}\boldsymbol{D}_2\boldsymbol{v}_2\right) = -\lambda\boldsymbol{H}^{-1}\boldsymbol{\mathcal{F}}_p. \tag{3.13}$$

Here, we have defined the matrices

$$\boldsymbol{H} = \boldsymbol{H}_{N_1} \otimes \boldsymbol{H}_{N_2}, \qquad \boldsymbol{D}_1 = \boldsymbol{D}_{N_1} \otimes \boldsymbol{I}_{N_2}, \qquad \boldsymbol{D}_2 = \boldsymbol{I}_{N_1} \otimes \boldsymbol{D}_{N_2}, \tag{3.14}$$

where $\boldsymbol{I}_{N_i}$, $\boldsymbol{H}_{N_i}$, and $\boldsymbol{D}_{N_i}$ are all matrices of size $(N_i+1)\times(N_i+1)$ with $\boldsymbol{I}_{N_i}$ being the identity matrix, and $\boldsymbol{H}_{N_i}$ and $\boldsymbol{D}_{N_i}$ being the 1-D SBP finite difference operators in the ξ_i direction. The vectors $\boldsymbol{v}_1$, $\boldsymbol{v}_2$, and $\boldsymbol{p}$ are the unknown velocities and pressures at the finite difference grid points. The diagonal matrices $\boldsymbol{J}$, $\frac{\partial \boldsymbol{\xi}_1}{\partial \boldsymbol{x}_1}$, $\frac{\partial \boldsymbol{\xi}_1}{\partial \boldsymbol{x}_2}$, $\frac{\partial \boldsymbol{\xi}_2}{\partial \boldsymbol{x}_1}$, and $\frac{\partial \boldsymbol{\xi}_2}{\partial \boldsymbol{x}_2}$ have the respective geometric factors evaluated at the finite difference grid points along their diagonal. For example letting J_{kl} denote the Jacobian determinant (or its approximation) at grid point (k, l) we define

$$\boldsymbol{J} = \operatorname{diag}\begin{bmatrix} J_{00} & J_{01} & \cdots & J_{0N_2} & J_{10} & \cdots & J_{N_1N_2} \end{bmatrix}, \tag{3.15}$$

where $\operatorname{diag}(\cdot)$ constructs a diagonal matrix from a given vector; the other diagonal matrices are defined similarly. For simplicity of the discussion we assume that the material parameters ρ and λ are constants in Ω.

Before stating the specific form for the penalty terms $\boldsymbol{\mathcal{F}}_{v_i}$ and $\boldsymbol{\mathcal{F}}_p$, we note the similarities between the SBP-SAT discretization and a DG method based on the integral form. Namely, if $\boldsymbol{H}$ is interpreted as an elemental mass matrix then the left-hand side of the (3.12)–(3.13) correspond to the body integral terms in (3.6)–(3.7). Similarly, the right-hand side terms correspond to the boundary integrals. Thus, the SBP-SAT discretization and the DG method have the same discrete structure. Additionally, what are commonly referred to as penalty terms in SBP-SAT finite difference methods are of the same form as the flux terms in DG methods. This is important because even though the methods are developed using different continuous representations (i.e., differential versus variational form), these similarities will allow stable coupling through the penalty and flux terms, see § 4.2.

The penalty terms in (3.12)–(3.13) are taken to be of the form

$$\boldsymbol{\mathcal{F}}_{v_i} = \left(\boldsymbol{e}_W \otimes \boldsymbol{\mathcal{F}}_{v_i}^W\right) + \left(\boldsymbol{e}_E \otimes \boldsymbol{\mathcal{F}}_{v_i}^E\right) + \left(\boldsymbol{\mathcal{F}}_{v_i}^S \otimes \boldsymbol{e}_S\right) + \left(\boldsymbol{\mathcal{F}}_{v_i}^N \otimes \boldsymbol{e}_N\right), \tag{3.16}$$

$$\boldsymbol{\mathcal{F}}_p = \left(\boldsymbol{e}_W \otimes \boldsymbol{\mathcal{F}}_p^W\right) + \left(\boldsymbol{e}_E \otimes \boldsymbol{\mathcal{F}}_p^E\right) + \left(\boldsymbol{\mathcal{F}}_p^S \otimes \boldsymbol{e}_S\right) + \left(\boldsymbol{\mathcal{F}}_p^N \otimes \boldsymbol{e}_N\right). \tag{3.17}$$

Here the subscripts and superscript W, E, S, and N are used to denote which side of the domain the penalty term correspond to. For instance, W and E correspond to the *west* and *east* sides of the domain with $\xi_1 = -1$ and $\xi_1 = 1$, respectively. Similarly, S and N correspond to the *south* and *north* with $\xi_2 = -1$ and $\xi_2 = 1$. The vectors $\boldsymbol{e}_W$ and $\boldsymbol{e}_E$ have length $N_1 + 1$ and are zero everywhere except the first and last entry, respectively, where they are 1, i.e.,

$$\boldsymbol{e}_W = \begin{bmatrix} 1 & 0 & \cdots & 0 \end{bmatrix}^T \quad \text{and} \quad \boldsymbol{e}_E = \begin{bmatrix} 0 & \cdots & 0 & 1 \end{bmatrix}^T; \tag{3.18}$$

the vectors $\boldsymbol{e}_S$ and $\boldsymbol{e}_N$ are defined analogously.

The vectors $\boldsymbol{\mathcal{F}}^W_{v_i}$ and $\boldsymbol{\mathcal{F}}^W_p$, of length N_2+1, are the actual penalty terms (or flux difference) along the west side. These vectors are taken to have the form

$$\boldsymbol{\mathcal{F}}^W_{v_i} = \boldsymbol{H}_2 \boldsymbol{S}_{JW} \boldsymbol{n}^W_i \left(\boldsymbol{p}^*_W - \boldsymbol{p}_W \right), \qquad \boldsymbol{\mathcal{F}}^W_p = \boldsymbol{H}_2 \boldsymbol{S}_{JW} \lambda \left(\boldsymbol{v}^*_W - \boldsymbol{v}_W \right). \tag{3.19}$$

Here $\boldsymbol{S}_{JW}$ and $\boldsymbol{n}^W_i$ are $(N_2+1) \times (N_2+1)$ diagonal matrices with diagonal elements corresponding to the surface Jacobian terms and outward pointing normals along the west face; see (3.8). The vectors $\boldsymbol{p}_W$ and $\boldsymbol{v}_W$, of length (N_2+1), are the pressure and normal components of velocity ($v = v_1 n_1 + v_2 n_2$), respectively, at grid points along the west face. Finally, the vectors $\boldsymbol{p}^*_W$ and $\boldsymbol{v}^*_W$ will be set based on the interface and/or boundary conditions for the block. In the case of interface conditions, these edge values will need to be set in a consistent manner across the interface. As noted above, these penalty terms correspond to the numerical fluxes in DG methods, and stability results through a judicious choice of penalty or flux.

Stability of the semi-discrete discretization, whether a pure multiblock SBP-SAT discretization or a coupled SBP-DG method, will be based on energy analysis. To do so we define the energy in a single SBP block as

$$E = \frac{\rho}{2} \boldsymbol{v}_1^T \boldsymbol{J} \boldsymbol{H} \boldsymbol{v}_1 + \frac{\rho}{2} \boldsymbol{v}_2^T \boldsymbol{J} \boldsymbol{H} \boldsymbol{v}_2 + \frac{1}{2\lambda} \boldsymbol{p}^T \boldsymbol{J} \boldsymbol{H} \boldsymbol{p}; \tag{3.20}$$

and define the total energy in the solution as

$$\mathcal{E} = \sum_{blocks} E. \tag{3.21}$$

Since the governing equations (3.1) are energy conservative with the free surface boundary condition, it is appropriate to use the following definition of discrete stability [4]:

Definition 3.1 (Energy Stability). *The semi-discrete discretization is said to be stable if*

$$\frac{d\mathcal{E}}{dt} \leq 0. \tag{3.22}$$

For a single SBP block, the energy dissipation rate is characterized by the following lemma:

Lemma 3.2. *The single SBP block discretization (3.12)–(3.13) with penalty terms (3.16)–(3.17) of the form of (3.19) has the energy dissipation rate*

$$\frac{dE}{dt} = \sum_{K=\{W,E,S,N\}} \mathcal{D}_K, \tag{3.23}$$

$$\mathcal{D}_K = - \boldsymbol{v}_K^T \boldsymbol{H}_K \boldsymbol{S}_{JK} \boldsymbol{p}^*_K + \boldsymbol{v}_K^T \boldsymbol{H}_K \boldsymbol{S}_{JK} \boldsymbol{p}_K - \left(\boldsymbol{v}^*_K \right)^T \boldsymbol{H}_K \boldsymbol{S}_{JK} \boldsymbol{p}_K, \tag{3.24}$$

where $\mathcal{D}_K$ is the dissipation rate along edge K of the block with $\boldsymbol{H}_W = \boldsymbol{H}_E = \boldsymbol{H}_2$ and $\boldsymbol{H}_S = \boldsymbol{H}_N = \boldsymbol{H}_1$.

Proof. See Appendix A. □ The implication of the lemma is that the energy dissipation rate for a single block is the sum of the dissipation rate for each of its edges. Thus, we can prove global semi-discrete energy stability by showing that energy is dissipation across every interface and boundary.

The crux of a stable coupling is then choosing $\boldsymbol{p}^*$ and $\boldsymbol{v}^*$ such that when (3.23) is summed over all blocks $d\mathcal{E}/dt \leq 0$. Before continuing on to present how the interface terms $\boldsymbol{p}^*$ and $\boldsymbol{v}^*$ are formulated when projection operators are used, we consider the form of the penalty terms take for an exterior boundary and when the interface is conforming (matching grid and SBP finite difference scheme across interface).

3.1. Exterior Boundary Treatment. Since the focus of this work is interface treatment, we will only consider the zero pressure boundary condition $p = 0$. Numerically, through the penalty terms, this enforcement is done through a linear combination of a central and upwind biased penalty; here by central we mean a penalty term that leads to no energy dissipation. If a block edge $K \in \{W, E, S, N\}$ is an outer boundary, the penalty terms are taken to be of the form

$$\boldsymbol{p}_K^* - \boldsymbol{p}_K = -\boldsymbol{p}_K, \qquad \boldsymbol{v}_K^* - \boldsymbol{v}_K = \alpha \frac{\boldsymbol{p}_K}{Z}, \tag{3.25}$$

where $Z = \sqrt{\rho/\lambda} > 0$ is the impedance of the material. Here the parameter $\alpha \geq 0$ has been introduced with $\alpha = 0$ being the central boundary treatment and $\alpha = 1$ being the fully upwind boundary treatment. The following lemma assures that the external boundary treatment is dissipative.

LEMMA 3.3. *If edge $K \in \{W, E, S, N\}$ of an SBP block is an exterior boundary with penalty terms of the form (3.25) then the energy dissipation rate for the edge is*

$$\mathcal{D}_K = -\frac{\alpha}{Z} \boldsymbol{p}_K^T \boldsymbol{H}_K \boldsymbol{S}_{JK} \boldsymbol{p}_K, \tag{3.26}$$

which is negative semidefinite if $\alpha \geq 0$.

Proof. Solving penalty term (3.25) for $\boldsymbol{p}_K^*$ and $\boldsymbol{v}_K^*$ gives

$$\boldsymbol{p}_K^* = 0, \qquad \boldsymbol{v}_K^* = \boldsymbol{v}_K + \alpha \frac{\boldsymbol{p}_K}{Z}, \tag{3.27}$$

and (3.26) follows immediately after substituting $\boldsymbol{p}_K^*$ and $\boldsymbol{v}_K^*$ into the edge dissipation rate (3.24). The negative semidefiniteness of (3.26) follows from the fact that $\boldsymbol{H}_K$ and $\boldsymbol{S}_{JK}$ are diagonal, positive definite matrices. □

3.2. Conforming Interface Treatment. We denote a block interface as conforming when the grid and the $\boldsymbol{H}$-norm are the same on both sides of the interface; the latter condition typically implies that the same SBP finite difference method is being used on both sides of the interface. The interface conditions are continuity of pressure and the normal component of velocity:

$$p^+ = p^-, \qquad v^+ = -v^-. \tag{3.28}$$

Here we have introduced the superscripts $+$ and $-$ to denote the two sides of the interface. Recall that v is the normal component of the velocity and thus the minus sign in the velocity condition is due to the fact that the normals are equal and opposite

on either side of the interface. For the minus side of the interface the penalty terms can then be written as a combination of the central and upwind penalties:

$$\boldsymbol{p}^* - \boldsymbol{p} = \boldsymbol{p}^* - \boldsymbol{p}^- = \frac{1}{2}\left(\boldsymbol{p}^+ - \boldsymbol{p}^-\right) + \alpha\frac{Z}{2}\left(\boldsymbol{v}^+ + \boldsymbol{v}^-\right), \tag{3.29}$$

$$\boldsymbol{v}^* - \boldsymbol{v} = \boldsymbol{v}^* - \boldsymbol{v}^- = -\frac{1}{2}\left(\boldsymbol{v}^+ + \boldsymbol{v}^-\right) - \alpha\frac{1}{2Z}\left(\boldsymbol{p}^+ - \boldsymbol{p}^-\right), \tag{3.30}$$

where stability results when $\alpha \geq 0$, and the central penalty (zero energy dissipation) is achieved when $\alpha = 0$ and the upwind penalty with $\alpha = 1$.

LEMMA 3.4. *Consider a single, conforming interface between two SBP blocks with penalty terms of the form (3.29)–(3.30). Let $\mathcal{D}^-$ and $\mathcal{D}^+$ be the energy dissipation rate along each side of the interface, then*

$$\begin{aligned}\mathcal{D}^- + \mathcal{D}^+ = & -\alpha\frac{Z}{2}\left(\boldsymbol{v}^- + \boldsymbol{v}^+\right)^T \boldsymbol{H}\boldsymbol{S}_J\left(\boldsymbol{v}^- + \boldsymbol{v}^+\right) \\ & -\alpha\frac{1}{2Z}\left(\boldsymbol{p}^- - \boldsymbol{p}^+\right)^T \boldsymbol{H}\boldsymbol{S}_J\left(\boldsymbol{p}^- - \boldsymbol{p}^+\right),\end{aligned} \tag{3.31}$$

is negative semidefinite for $\alpha \geq 0$.

Proof. Solving (3.29)–(3.30) for $\boldsymbol{p}^*$ and $\boldsymbol{v}^*$ on the minus side of the interface gives

$$\boldsymbol{p}^* = \frac{1}{2}\left(\boldsymbol{p}^+ + \boldsymbol{p}^-\right) + \alpha\frac{Z}{2}\left(\boldsymbol{v}^+ + \boldsymbol{v}^-\right), \tag{3.32}$$

$$\boldsymbol{v}^* = \frac{1}{2}\left(\boldsymbol{v}^- - \boldsymbol{v}^+\right) + \alpha\frac{1}{2Z}\left(\boldsymbol{p}^- - \boldsymbol{p}^+\right). \tag{3.33}$$

Substituting $\boldsymbol{p}^*$ and $\boldsymbol{v}^*$ into (3.24) on the minus side of the interface results in (after some simplification)

$$\begin{aligned}\mathcal{D}^- = & -\frac{1}{2}\left(\boldsymbol{v}^-\right)^T \boldsymbol{H}\boldsymbol{S}_J\boldsymbol{p}^+ - \alpha\frac{Z}{2}\left(\boldsymbol{v}^-\right)^T \boldsymbol{H}\boldsymbol{S}_J\left(\boldsymbol{v}^+ + \boldsymbol{v}^-\right) \\ & +\frac{1}{2}\left(\boldsymbol{p}^-\right)^T \boldsymbol{H}\boldsymbol{S}_J\boldsymbol{v}^+ + \alpha\frac{1}{2Z}\left(\boldsymbol{p}^-\right)^T \boldsymbol{H}\boldsymbol{S}_J\left(\boldsymbol{p}^+ - \boldsymbol{p}^-\right).\end{aligned} \tag{3.34}$$

A similar calculation for the plus side of the interface gives

$$\begin{aligned}\mathcal{D}^+ = & -\frac{1}{2}\left(\boldsymbol{v}^+\right)^T \boldsymbol{H}\boldsymbol{S}_J\boldsymbol{p}^- - \alpha\frac{Z}{2}\left(\boldsymbol{v}^+\right)^T \boldsymbol{H}\boldsymbol{S}_J\left(\boldsymbol{v}^- + \boldsymbol{v}^+\right) \\ & +\frac{1}{2}\left(\boldsymbol{p}^+\right)^T \boldsymbol{H}\boldsymbol{S}_J\boldsymbol{v}^- + \alpha\frac{1}{2Z}\left(\boldsymbol{p}^+\right)^T \boldsymbol{H}\boldsymbol{S}_J\left(\boldsymbol{p}^- - \boldsymbol{p}^+\right).\end{aligned} \tag{3.35}$$

Edge energy dissipation (3.31) then follows since $\boldsymbol{H}$ and $\boldsymbol{S}_J$ are diagonal matrices. Similarly, the negative semidefiniteness of (3.31) when $\alpha \geq 0$ follows from the diagonal, positive definiteness of $\boldsymbol{H}$ and $\boldsymbol{S}_J$. ∎

4. General interface treatment. Our discussion of more general interfaces begins with the coupling of two SBP finite difference blocks that conform at the block level (i.e., no hanging multiblock nodes). Throughout we assume that both blocks have the same continuous coordinate transform along the interface. For example, consider the case shown in Fig. 1.1 (left panel), where we assume that the block on the right side of the interface has been transformed with $x_1^+(\xi_1, \xi_2)$ and $x_2^+(\xi_1, \xi_2)$, and similarly the block on the left side has been transformed with $x_1^-(\xi_1, \xi_2)$ and $x_2^-(\xi_1, \xi_2)$. With this notation, both blocks see the same transform along the interface if $x_1^+(-1, \xi) =$

$x_1^-(1,\xi)$ and $x_2^+(-1,\xi) = x_2^-(1,\xi)$, where for simplicity we have assumed that the west face of the right block is connected to the east face of the left block. The glue grid is then parameterized by the variable $-1 \leq \eta \leq 1$. Note that we assume nothing about how many grid points are along this interface, only that they conform at the continuous level.

The core idea behind the non-conforming interface treatment is that the penalty terms are computed on a *glue grid* between the two domains. An example glue grid between two finite difference methods is shown in Fig. 2.2. As can be seen, the glue grid between the two domains is defined so that the grid points are nested with the glue grid interval boundaries.

To move values between the finite difference grid and the glue grid the previously defined projection operators are used. Namely the operators $\boldsymbol{P}_{f2g}^-$ and $\boldsymbol{P}_{f2g}^+$ move values from the grid on the minus and plus sides of the interface to the glue grid and $\boldsymbol{P}_{g2f}^-$ and $\boldsymbol{P}_{g2f}^+$ from the glue grid to the minus and plus side finite difference grids. We will see that since at the discrete level both sides of the interface may sample the geometry and metric terms differently, these geometry differences, specifically the surface Jacobian, must be taken into account in the projection to ensure discrete stability. To do this we project the square root of the surface Jacobians along with the grid values to the glue grid; since the surface Jacobian matrices $\boldsymbol{S}_J^\pm$ are positive, diagonal matrices the square root of these matrices are trivial to compute. Hence, the values that we work with on the glue grid are

$$\bar{\boldsymbol{p}}^\pm = \boldsymbol{P}_{f2g}^\pm \left(\boldsymbol{S}_J^\pm\right)^{1/2} \boldsymbol{p}^\pm, \qquad \bar{\boldsymbol{v}}^\pm = \boldsymbol{P}_{f2g}^\pm \left(\boldsymbol{S}_J^\pm\right)^{1/2} \boldsymbol{v}^\pm; \tag{4.1}$$

we note that values on the glue grid are always scaled by square root of the surface Jacobian. Here, the vectors $\boldsymbol{p}^\pm$ and $\boldsymbol{v}^\pm$ refer only to pressure values and normal component of velocity along the interface of interest.

With this notation, the penalty terms along a non-conforming interface are:

$$\boldsymbol{p}^* - \boldsymbol{p}^- = \left(\boldsymbol{S}_J^-\right)^{-1/2} \boldsymbol{P}_{g2f}^- \left(\bar{\boldsymbol{p}}^* - \bar{\boldsymbol{p}}^-\right) + \frac{1}{2}\left[\left(\boldsymbol{S}_J^-\right)^{-1/2} \boldsymbol{P}_{g2f}^- \bar{\boldsymbol{p}}^- - \boldsymbol{p}^-\right], \tag{4.2}$$

$$\boldsymbol{v}^* - \boldsymbol{v}^- = \left(\boldsymbol{S}_J^-\right)^{-1/2} \boldsymbol{P}_{g2f}^- \left(\bar{\boldsymbol{v}}^* - \bar{\boldsymbol{v}}^-\right) + \frac{1}{2}\left[\left(\boldsymbol{S}_J^-\right)^{-1/2} \boldsymbol{P}_{g2f}^- \bar{\boldsymbol{v}}^- - \boldsymbol{v}^-\right], \tag{4.3}$$

where $\bar{\boldsymbol{p}}^* - \bar{\boldsymbol{p}}^-$ and $\bar{\boldsymbol{v}}^* - \bar{\boldsymbol{v}}^-$ are defined by (3.29)–(3.30) using the values $\bar{\boldsymbol{p}}^\pm$ and $\bar{\boldsymbol{v}}^\pm$ for $\boldsymbol{p}^\pm$ and $\boldsymbol{v}^\pm$, respectively. As in the conforming case, the parameter $\alpha \geq 0$ controls the central versus upwind biasness of the scheme. The second term on the right-hand-side of (4.2) (and (4.3)) is a projection error which arises because $\boldsymbol{P}_{g2f}\boldsymbol{P}_{f2g}$ is not an identity operation.

An important implication of the penalty terms (4.2)–(4.3) is that the projection operations for the two sides are independent of the scheme on either side of the interface and the underlying representation of the geometry. This later fact means that the geometry does not need to be build into the projection operation. Also note that if the interface is conforming, the conforming penalties (3.29)–(3.30) are equivalent to the non-conforming penalty terms (4.2)–(4.3) if the projection matrices are taken to be the identity matrix: $\boldsymbol{P}_{f2g}^\pm = \boldsymbol{P}_{g2f}^\pm = \boldsymbol{I}$.

We can now state the first major result of the paper:

THEOREM 4.1. *Consider a single, non-conforming interface between two SBP blocks with penalty terms of the form (4.2)–(4.3). Let $\mathcal{D}^-$ and $\mathcal{D}^+$ be the energy*

dissipation rate along each side of the interface, then

$$\begin{aligned}\mathcal{D}^- + \mathcal{D}^+ = & -\alpha\frac{Z}{2}\left(\bar{\boldsymbol{v}}^- + \bar{\boldsymbol{v}}^+\right)^T \boldsymbol{M}\left(\bar{\boldsymbol{v}}^- + \bar{\boldsymbol{v}}^+\right) \\ & -\alpha\frac{1}{2Z}\left(\bar{\boldsymbol{p}}^- - \bar{\boldsymbol{p}}^+\right)^T \boldsymbol{M}\left(\bar{\boldsymbol{p}}^- - \bar{\boldsymbol{p}}^+\right), \end{aligned} \tag{4.4}$$

which is negative semidefinite for $\alpha \geq 0$.

Proof. Solving (4.2)–(4.3) for $\boldsymbol{p}^*$ and $\boldsymbol{v}^*$ on the minus side of the interface gives

$$\boldsymbol{p}^* = \left(\boldsymbol{S}_J^-\right)^{-1/2}\boldsymbol{P}_{g2f}^-\left(\bar{\boldsymbol{p}}^* - \bar{\boldsymbol{p}}^-\right) + \frac{1}{2}\left[\left(\boldsymbol{S}_J^-\right)^{-1/2}\boldsymbol{P}_{g2f}^-\bar{\boldsymbol{p}}^- + \boldsymbol{p}^-\right], \tag{4.5}$$

$$\boldsymbol{v}^* = \left(\boldsymbol{S}_J^-\right)^{-1/2}\boldsymbol{P}_{g2f}^-\left(\bar{\boldsymbol{v}}^* - \bar{\boldsymbol{v}}^-\right) + \frac{1}{2}\left[\left(\boldsymbol{S}_J^-\right)^{-1/2}\boldsymbol{P}_{g2f}^-\bar{\boldsymbol{v}}^- + \boldsymbol{v}^-\right]. \tag{4.6}$$

Substituting $\boldsymbol{p}^*$ and $\boldsymbol{v}^*$ into (3.24) on the minus side of the interface results in (after some simplification)

$$\begin{aligned}\mathcal{D}^- = & -\left(\boldsymbol{v}^-\right)^T\boldsymbol{H}^-\left(\boldsymbol{S}_J^-\right)^{1/2}\boldsymbol{P}_{g2f}^-\left(\bar{\boldsymbol{p}}^* - \bar{\boldsymbol{p}}^-\right) - \frac{1}{2}\left(\boldsymbol{v}^-\right)^T\boldsymbol{H}^-\left(\boldsymbol{S}_J^-\right)^{1/2}\boldsymbol{P}_{g2f}^-\bar{\boldsymbol{p}}^- \\ & -\left(\boldsymbol{p}^-\right)^T\boldsymbol{H}^-\left(\boldsymbol{S}_j^-\right)^{1/2}\boldsymbol{P}_{g2f}^-\left(\bar{\boldsymbol{v}}^* - \bar{\boldsymbol{v}}^-\right) - \frac{1}{2}\left(\boldsymbol{p}^-\right)^T\boldsymbol{H}^-\left(\boldsymbol{S}_j^-\right)^{1/2}\boldsymbol{P}_{g2f}^-\bar{\boldsymbol{v}}^-. \end{aligned} \tag{4.7}$$

Using property (2.4) of the projection operator the energy dissipation on the minus side of the interface is

$$\begin{aligned}\mathcal{D}^- = & -\left(\boldsymbol{v}^-\right)^T\left(\boldsymbol{S}_J^-\right)^{1/2}\left(\boldsymbol{P}_{f2g}^-\right)^T\boldsymbol{M}\left(\bar{\boldsymbol{p}}^* - \bar{\boldsymbol{p}}^-\right) - \frac{1}{2}\left(\boldsymbol{v}^-\right)^T\left(\boldsymbol{S}_J^-\right)^{1/2}\left(\boldsymbol{P}_{f2g}^-\right)^T\boldsymbol{M}\bar{\boldsymbol{p}}^- \\ & -\left(\boldsymbol{p}^-\right)^T\left(\boldsymbol{S}_j^-\right)^{1/2}\left(\boldsymbol{P}_{f2g}^-\right)^T\boldsymbol{M}\left(\bar{\boldsymbol{v}}^* - \bar{\boldsymbol{v}}^-\right) - \frac{1}{2}\left(\boldsymbol{p}^-\right)^T\left(\boldsymbol{S}_j^-\right)^{1/2}\left(\boldsymbol{P}_{f2g}^-\right)^T\boldsymbol{M}\bar{\boldsymbol{v}}^- \\ = & -\left(\bar{\boldsymbol{v}}^-\right)^T\boldsymbol{M}\bar{\boldsymbol{p}}^* + \left(\bar{\boldsymbol{v}}^-\right)^T\boldsymbol{M}\bar{\boldsymbol{p}}^- - \left(\bar{\boldsymbol{p}}^-\right)^T\boldsymbol{M}\bar{\boldsymbol{v}}^* \end{aligned} \tag{4.8}$$

where we have used that $\boldsymbol{H}$ and $\boldsymbol{S}_J$ commute since they are diagonal as well as the definitions of $\bar{\boldsymbol{p}}^-$ and $\bar{\boldsymbol{v}}^-$; see (4.1). Solving (3.29) and (3.30), evaluated with $\bar{\boldsymbol{p}}^\pm$ and $\bar{\boldsymbol{v}}^\pm$, for $\bar{\boldsymbol{p}}^*$ and $\bar{\boldsymbol{v}}^*$ and substituting these values into (4.8) gives (after minor algebraic manipulations)

$$\begin{aligned}\mathcal{D}^- = & -\frac{1}{2}\left(\bar{\boldsymbol{v}}^-\right)^T\boldsymbol{M}\bar{\boldsymbol{p}}^+ - \alpha\frac{Z}{2}\left(\bar{\boldsymbol{v}}^-\right)^T\boldsymbol{M}\left(\bar{\boldsymbol{v}}^+ + \bar{\boldsymbol{v}}^-\right) \\ & +\frac{1}{2}\left(\bar{\boldsymbol{p}}^-\right)^T\boldsymbol{M}\bar{\boldsymbol{v}}^+ + \alpha\frac{1}{2Z}\left(\bar{\boldsymbol{p}}^-\right)^T\boldsymbol{M}\left(\bar{\boldsymbol{p}}^+ - \bar{\boldsymbol{p}}^-\right). \end{aligned} \tag{4.9}$$

A similar calculation for the plus side of the interface gives

$$\begin{aligned}\mathcal{D}^+ = & -\frac{1}{2}\left(\bar{\boldsymbol{v}}^+\right)^T\boldsymbol{M}\bar{\boldsymbol{p}}^- - \alpha\frac{Z}{2}\left(\bar{\boldsymbol{v}}^+\right)^T\boldsymbol{M}\left(\bar{\boldsymbol{v}}^- + \bar{\boldsymbol{v}}^+\right) \\ & +\frac{1}{2}\left(\bar{\boldsymbol{p}}^+\right)^T\boldsymbol{M}\bar{\boldsymbol{v}}^- + \alpha\frac{1}{2Z}\left(\bar{\boldsymbol{p}}^+\right)^T\boldsymbol{M}\left(\bar{\boldsymbol{p}}^- - \bar{\boldsymbol{p}}^+\right). \end{aligned} \tag{4.10}$$

Summing (4.9) and (4.10) then gives (4.4). Similarly, the negative semidefiniteness of (4.4) when $\alpha \geq 0$ follows from the positive definiteness of $\boldsymbol{M}$. ∎

Comparison with Mattsson and Carpenter [11] Interpolation Operators. As noted above, Mattsson and Carpenter have previously proposed a set of SBP-compatible operators for coupling conforming (at the block level) SBP finite difference methods with a fixed refinement ratio [11]. These operators, which Mattsson and Carpenter denote as $\boldsymbol{I}_{F2C}$ and $\boldsymbol{I}_{C2F}$ with $F2C$ and $C2F$ denoting fine to coarse and vice versa, move a solution all the way from one finite difference grid to the next. Thus, an important difference with the projection operators we employ here is that there is an intermediate glue grid which allows the projection operators to be defined independent of the coupling; Mattsson and Carpenter's operators depend on both the refinement ratio and SBP operator on either side of the interface. Additionally, Mattsson and Carpenter had to introduce additional constraints in order to ensure stability when an upwind bias penalties/numerical fluxes are used; see equation (16) of Mattsson and Carpenter [11]. In their paper, Mattsson and Carpenter note that they were unable to construct operators which always satisfy these constraints and for some cases dissipation was introduced to stabilize the method. In this work the use of the glue grid allows us to overcome these extra constraints on the operators as well as the need to introduce dissipation on the interface; note that in the method we propose here there is dissipation on the interface and it is controlled by the upwind parameter α. Finally, it should be noted that the operators $\boldsymbol{I}_{F2C} = \boldsymbol{P}^{+}_{g2f}\boldsymbol{P}^{-}_{f2g}$ and $\boldsymbol{I}_{C2F} = \boldsymbol{P}^{-}_{g2f}\boldsymbol{P}^{+}_{f2g}$ satisfy stability constraints (15) and the accuracy conditions of Definition 2.4 of Mattsson and Carpenter [11].

4.1. Many-to-many interfaces. We now move on to the case when several finite difference blocks are coupled together along a single interface. As will be seen, the treatment for this case is identical to the one-to-one interface case except that the surface Jacobians of the blocks along the coupling interface must be scaled to put them into the glue grid space. An example of the sort of coupling considered is shown in the center panel of Fig. 1.1 where we are interested in the treatment of the black interface; without loss of generality we assume that the interface occurs in the ξ_2 direction for all blocks.

As noted above, we parameterize the glue space with a variable $-1 \leq \eta \leq 1$. We let N^- be then number of blocks along the minus side of the interface and each block k, $1 \leq k \leq N^-$, overlaps the glue interface over $\beta^{-(k-1)} \leq \eta \leq \beta^{-(k)}$ with $\beta^{-(0)} = -1$ and $\beta^{-(N^-)} = 1$. We then define the affine interface transform to take each block interface (which run from $-1 \leq \xi \leq 1$) to the appropriate portion of the glue interface:

$$\eta^{-(k)} = \frac{\beta^{-(k-1)}(1-\xi_2^{-(k)}) + \beta^{-(k)}(1+\xi_2^{-(k)})}{2}, \tag{4.11}$$

$$\xi_2^{-(k)} = \frac{(\eta^{-(k)} - \beta^{-(k)}) + (\eta^{-(k)} - \beta^{-(k-1)})}{\beta^{-(k)} - \beta^{-(k-1)}}, \tag{4.12}$$

where $1 \leq k \leq N^-$ denotes which of the blocks along this side of the interface we are considering. Since these are affine transforms, their effect on the surface Jacobian (see (3.8)) will be a constant scaling of

$$\frac{\partial \xi_2^{-(k)}}{\partial \eta^{-(k)}} = \frac{2}{\beta^{-(k)} - \beta^{-(k-1)}} = \frac{1}{\Delta^{-(k)}}. \tag{4.13}$$

Here $\Delta^{-(k)}$ is the fraction of the interface which intersects block k. As similar construction is used for the N^+ blocks on the other side of the interface.

Projections to the glue and the penalty terms for each block are then defined as in (4.1), (4.2), and (4.3) except with the surface Jacobians scaled by $1/\Delta^{\pm(k)}$:

$$\bar{\boldsymbol{p}}^{\pm(k)} = \boldsymbol{P}_{f2g}^{\pm(k)} \left(\frac{\boldsymbol{S}_J^{\pm(k)}}{\Delta^{\pm(k)}} \right)^{1/2} \boldsymbol{p}^{\pm(k)}, \qquad \bar{\boldsymbol{v}}^{\pm(k)} = \boldsymbol{P}_{f2g}^{\pm(k)} \left(\frac{\boldsymbol{S}_J^{\pm(k)}}{\Delta^{\pm(k)}} \right)^{1/2} \boldsymbol{v}^{\pm(k)}, \tag{4.14}$$

$$\begin{aligned} \boldsymbol{p}^{*(k)} - \boldsymbol{p}^{-(k)} = & \left(\frac{\boldsymbol{S}_J^{-(k)}}{\Delta^{-(k)}} \right)^{-1/2} \boldsymbol{P}_{g2f}^{-(k)} \left(\bar{\boldsymbol{p}}^{*(k)} - \bar{\boldsymbol{p}}^{-(k)} \right) \\ & + \frac{1}{2} \left[\left(\frac{\boldsymbol{S}_J^{-(k)}}{\Delta^{-(k)}} \right)^{-1/2} \boldsymbol{P}_{g2f}^{-(k)} \bar{\boldsymbol{p}}^{-(k)} - \boldsymbol{p}^{-(k)} \right], \end{aligned} \tag{4.15}$$

$$\begin{aligned} \boldsymbol{v}^{*(k)} - \boldsymbol{v}^{-(k)} = & \left(\frac{\boldsymbol{S}_J^{-(k)}}{\Delta^{-(k)}} \right)^{-1/2} \boldsymbol{P}_{g2f}^{-(k)} \left(\bar{\boldsymbol{v}}^{*(k)} - \bar{\boldsymbol{v}}^{-(k)} \right) \\ & + \frac{1}{2} \left[\left(\frac{\boldsymbol{S}_J^{-(k)}}{\Delta^{-(k)}} \right)^{-1/2} \boldsymbol{P}_{g2f}^{-(k)} \bar{\boldsymbol{v}}^{-(k)} - \boldsymbol{v}^{-(k)} \right]. \end{aligned} \tag{4.16}$$

Before going on to state the edge dissipation rates, we note that the change of variables (4.11)–(4.12) requires a slight modification to the H-compatible definition (2.2), namely we now use the definition

$$\Delta^{-(k)} \left(\boldsymbol{P}_{g2f}^{-(k)} \right)^T \boldsymbol{H}^{-(k)} = \boldsymbol{M}^{-(k)} \boldsymbol{P}_{f2g}^{-(k)}. \tag{4.17}$$

Here, $\boldsymbol{H}^{-(k)}$ is the 1-D SBP norm matrix for a grid from -1 to 1 and $\boldsymbol{M}^{-(k)}$ is the mass matrix for the portion of the glue grid running from $\beta^{-(k-1)}$ to $\beta^{-(k)}$. The difference between the domains of $\boldsymbol{H}^{-(k)}$ and $\boldsymbol{M}^{-(k)}$ gives rise to the $\Delta^{-(k)}$ in the compatibility condition.

We define the solutions vectors on the glue as

$$\bar{\boldsymbol{v}}^{\pm} = \begin{bmatrix} \bar{\boldsymbol{v}}^{\pm(1)} \\ \vdots \\ \bar{\boldsymbol{v}}^{\pm(N^{\pm})} \end{bmatrix}, \qquad \bar{\boldsymbol{p}}^{\pm} = \begin{bmatrix} \bar{\boldsymbol{p}}^{\pm(1)} \\ \vdots \\ \bar{\boldsymbol{p}}^{\pm(N^{\pm})} \end{bmatrix} \tag{4.18}$$

and the block diagonal glue mass matrix

$$\boldsymbol{M} = \begin{bmatrix} \boldsymbol{M}^{-(1)} & & \\ & \ddots & \\ & & \boldsymbol{M}^{-(N^-)} \end{bmatrix} = \begin{bmatrix} \boldsymbol{M}^{+(1)} & & \\ & \ddots & \\ & & \boldsymbol{M}^{+(N^+)} \end{bmatrix}, \tag{4.19}$$

where we note that it is equivalent to define $\boldsymbol{M}$ from either the plus or minus side block mass matrices since they integrate the same space of function after stacking. With these definitions, we now have the following theorem which guarantees stability of the interface treatment.

THEOREM 4.2. *Consider a single, non-conforming interface with N^- and N^+ SBP blocks on either side of the interface with penalty terms of the form (4.15)–(4.16).*

Let $\mathcal{D}^-$ and $\mathcal{D}^+$ be the energy dissipation rate along each side of the interface, then

$$\begin{aligned}\mathcal{D}^- + \mathcal{D}^+ = & -\alpha\frac{Z}{2}\left(\bar{\boldsymbol{v}}^- + \bar{\boldsymbol{v}}^+\right)^T \boldsymbol{M}\left(\bar{\boldsymbol{v}}^- + \bar{\boldsymbol{v}}^+\right) \\ & -\alpha\frac{1}{2Z}\left(\bar{\boldsymbol{p}}^- - \bar{\boldsymbol{p}}^+\right)^T \boldsymbol{M}\left(\bar{\boldsymbol{p}}^- - \bar{\boldsymbol{p}}^+\right),\end{aligned} \tag{4.20}$$

is the negative semidefinite for $\alpha \geq 0$.

Proof. Solving the penalty terms for $\boldsymbol{p}^{*(k)}$ and $\boldsymbol{v}^{*(k)}$ and substituting into (3.24) gives (after a calculation similar to (4.8)) the edge dissipation rate

$$\begin{aligned}\mathcal{D}^{-(k)} = & -\left(\bar{\boldsymbol{v}}^{-(k)}\right)^T \boldsymbol{M}^{-(k)}\bar{\boldsymbol{p}}^{*(k)} + \left(\bar{\boldsymbol{v}}^{-(k)}\right)^T \boldsymbol{M}^{-(k)}\bar{\boldsymbol{p}}^{-(k)} \\ & -\left(\bar{\boldsymbol{p}}^{-(k)}\right)^T \boldsymbol{M}^{-(k)}\bar{\boldsymbol{v}}^{*(k)}.\end{aligned} \tag{4.21}$$

Defining the vectors

$$\bar{\boldsymbol{v}}^* = \begin{bmatrix} \bar{\boldsymbol{v}}^{*(1)} \\ \vdots \\ \bar{\boldsymbol{v}}^{*(N^-)} \end{bmatrix}, \qquad \bar{\boldsymbol{p}}^* = \begin{bmatrix} \bar{\boldsymbol{p}}^{*(1)} \\ \vdots \\ \bar{\boldsymbol{p}}^{*(N^-)} \end{bmatrix} \tag{4.22}$$

the sum of the contributions from all the blocks on the minus side of the interface can be written as

$$\mathcal{D}^- = \sum_{k=1}^{N^-} \mathcal{D}^{-(k)} = -\left(\bar{\boldsymbol{v}}^-\right)^T \boldsymbol{M}\bar{\boldsymbol{p}}^* + \left(\bar{\boldsymbol{v}}^-\right)^T \boldsymbol{M}\bar{\boldsymbol{p}} - \left(\bar{\boldsymbol{p}}^-\right)^T \boldsymbol{M}\bar{\boldsymbol{v}}^*. \tag{4.23}$$

Since this equation is identical to (4.9), the remainder of the proof is identical to that of Theorem 4.1. ∎

4.2. Connecting with discontinuous Galerkin methods. Besides allowing for the stable coupling of general finite difference grids, the projection operators defined above can also be used to couple finite difference methods with numerical methods in variational form. To demonstrate this we consider the coupling of SBP finite difference methods with a curvilinear, triangle based DG method. We begin by introducing a triangular, curved element DG method and then proceed to view each DG element as a small SBP finite difference block, which leads immediately to a stable coupling between the methods. Though we use one particular DG method, coupling with other formulations is possible as the coupling is purely done at the numerical flux level so any scheme that gives rise to similar interface terms will be stable. In what follows we only highlight the essential parts of the DG formulation that are necessary to couple it with SBP methods and for a more complete description of DG the interested reader is directed to, for instance, Hesthaven and Warburton [5].

To introduce the DG method we start with the variational form of the governing equations (3.6)-(3.7) for a DG element Ω_e whose reference element is $\tilde{\Omega}$. For the examples in this paper, we use a triangular reference element. Applying integration by parts to the conservation of momentum equation (3.6) in order to move the spatial derivatives from the solution p to the test functions w_i we get conservation of momentum in the form

$$\int_{\tilde{\Omega}} \left[w_i \rho J \frac{\partial v_i}{\partial t} - \frac{\partial w_i}{\partial \xi_1} J \frac{\partial \xi_1}{\partial x_i} p - \frac{\partial w_i}{\partial \xi_2} J \frac{\partial \xi_2}{\partial x_i} p \right] dA = -\int_{\partial\tilde{\Omega}} w_i S_J n_i p^* ds. \tag{4.24}$$

Notice that in the right-hand side of (4.24) depends only on p^*, i.e., the value which will become the numerical flux. The form of (4.24) with the derivative on the test function and (3.7) with the derivative on the trial function is sometimes referred to as the skew-symmetric form of the variational equations.

Discretizing the variational forms (4.24) and (3.7) in space using the DG method gives rise to the following semi-discretization on each element:

$$\rho \boldsymbol{M}_J \frac{d\boldsymbol{v}_i}{dt} = \boldsymbol{D}_1^T \boldsymbol{M}_{1i} \boldsymbol{p} + \boldsymbol{D}_2^T \boldsymbol{M}_{2i} \boldsymbol{p} - \sum_{K=1}^{3} \boldsymbol{L}_K^T \boldsymbol{P}_{bc}^T \boldsymbol{n}_{iK} \boldsymbol{\Omega}_{bc} \boldsymbol{S}_{JK} \boldsymbol{p}_K^*, \tag{4.25}$$

$$\begin{aligned} \boldsymbol{M}_J \frac{d\boldsymbol{p}}{dt} = & -\lambda \left(\boldsymbol{M}_{11} \boldsymbol{D}_1 \boldsymbol{v}_1 + \boldsymbol{M}_{21} \boldsymbol{D}_2 \boldsymbol{v}_1 + \boldsymbol{M}_{12} \boldsymbol{D}_1 \boldsymbol{v}_2 + \boldsymbol{M}_{22} \boldsymbol{D}_2 \boldsymbol{v}_2 \right) \\ & - \sum_{K=1}^{3} \lambda \boldsymbol{L}_K^T \boldsymbol{P}_{bc}^T \boldsymbol{\Omega}_{bc} \boldsymbol{S}_{JK} \left(\boldsymbol{v}_K^* - \boldsymbol{v}_K^- \right), \end{aligned} \tag{4.26}$$

where the vector $\boldsymbol{v}_K^-$ is the normal component of velocity along edge K of the element evaluated at the cubature points:

$$\boldsymbol{v}_K^- = \boldsymbol{n}_{1K} \boldsymbol{P}_{bc} \boldsymbol{L}_K \boldsymbol{v}_1 + \boldsymbol{n}_{2K} \boldsymbol{P}_{bc} \boldsymbol{L}_K \boldsymbol{v}_2. \tag{4.27}$$

Here $\boldsymbol{L}_K$ takes the volume terms to edge K of the element and $\boldsymbol{L}_K^T$ takes edge K terms to the volume; this is similar to the behavior of $\boldsymbol{e}_{W/E} \otimes \boldsymbol{I}$ and $\boldsymbol{I} \otimes \boldsymbol{e}_{N/S}$ in the SBP method. Also as in the SBP method, $\boldsymbol{D}_1$ and $\boldsymbol{D}_2$ are the reference element differentiation matrices for the two reference coordinate directions. Since we will be using curved triangular elements, integration is done using a cubature in the volume and quadratures along the edges of the elements. Thus we introduce the projection matrices $\boldsymbol{P}_c$ and $\boldsymbol{P}_{bc}$ that project from the volume and edge approximations to the volume and edge cubature points, respectively. At the cubature locations, the matrices $\boldsymbol{\Omega}_c$ and $\boldsymbol{\Omega}_{bc}$ are diagonal matrices of the integration weights for the volume and an edge, respectively. To ensure stability of the method, we will assume that the quadrature is such that $\boldsymbol{\Omega}_{bc}$ is positive definite. (Note, positive definiteness of $\boldsymbol{\Omega}_c$ is not required for stability.) The element mass matrices in the discretization are defined as

$$\boldsymbol{M}_J = \boldsymbol{P}_c^T \boldsymbol{\Omega}_c \boldsymbol{J} \boldsymbol{P}_c, \quad \boldsymbol{M}_{ij} = \boldsymbol{P}_c^T \boldsymbol{\Omega}_c \boldsymbol{J} \frac{\boldsymbol{\partial \xi_i}}{\boldsymbol{\partial x_j}} \boldsymbol{P}_c. \tag{4.28}$$

Here the diagonal matrices $\boldsymbol{J}$ and $\frac{\boldsymbol{\partial \xi_i}}{\boldsymbol{\partial x_j}}$ are, respectively, the Jacobian determinant and metric derivatives defined at the cubature points. The diagonal matrices $\boldsymbol{S}_{JK}$ and $\boldsymbol{n}_{iK}$ are the surface Jacobian and the components of the unit normal for edge K, respectively, defined at the cubature points.

Defining the energy in a DG element as

$$E = \frac{\rho}{2} \boldsymbol{v}_1^T \boldsymbol{M}_J \boldsymbol{v}_1 + \frac{\rho}{2} \boldsymbol{v}_2^T \boldsymbol{M}_J \boldsymbol{v}_2 + \frac{1}{2\lambda} \boldsymbol{p}^T \boldsymbol{M}_J \boldsymbol{p} \tag{4.29}$$

as well as the edge projected pressures

$$\boldsymbol{p}_K^- = \boldsymbol{P}_{bc} \boldsymbol{L}_K \boldsymbol{p}, \tag{4.30}$$

the energy dissipation rate for a single DG element can be characterized by the following lemma.

LEMMA 4.3. *The single DG block discretization (4.25)–(4.26) has the energy dissipation rate*

$$\frac{dE}{dt} = \sum_{K=1}^{3} \mathcal{D}_K, \tag{4.31}$$

$$\mathcal{D}_K = -\left(\boldsymbol{v}_K^-\right)^T \boldsymbol{\Omega}_{bc} \boldsymbol{S}_{JK} \boldsymbol{p}_K^* - \left(\boldsymbol{p}_K^-\right)^T \boldsymbol{\Omega}_{bc} \boldsymbol{S}_{JK} \left(\boldsymbol{v}_K^* - \boldsymbol{v}_K^-\right), \tag{4.32}$$

with energy as defined in (4.29)

Proof. Equation (4.32) follows directly by inserting (4.25)–(4.26) into the time derivative of (4.29)

$$\frac{dE}{dt} = \rho \boldsymbol{v}_1^T \boldsymbol{M}_J \frac{d\boldsymbol{v}_1}{dt} + \rho \boldsymbol{v}_2^T \boldsymbol{M}_J \frac{d\boldsymbol{v}_2}{dt} + \frac{1}{\lambda} \boldsymbol{p}^T \boldsymbol{M}_J \frac{d\boldsymbol{p}}{dt}, \tag{4.33}$$

and simplifying using the definition of the edge projected pressures (4.30) and normal velocity (4.27). The negative semidefiniteness of (4.32) follows from the fact that $\boldsymbol{\Omega}_{bc}$ and $\boldsymbol{S}_{JK}$ are diagonal, positive definite matrices. ∎

4.2.1. Boundary and DG-to-DG numerical flux. When an edge occurs on a boundary, the numerical flux is taken to be

$$\boldsymbol{p}_K^* = 0, \qquad \boldsymbol{v}_K^* - \boldsymbol{v}_K = \alpha \frac{\boldsymbol{p}_K}{Z}. \tag{4.34}$$

Similarly, the numerical flux between two DG elements is taken to be

$$\boldsymbol{p}^* = \frac{1}{2}\left(\boldsymbol{p}^+ + \boldsymbol{p}^-\right) + \alpha \frac{Z}{2}\left(\boldsymbol{v}^+ + \boldsymbol{v}^-\right), \tag{4.35}$$

$$\boldsymbol{v}^* - \boldsymbol{v}^- = -\frac{1}{2}\left(\boldsymbol{v}^+ + \boldsymbol{v}^-\right) - \alpha \frac{1}{2Z}\left(\boldsymbol{p}^+ - \boldsymbol{p}^-\right). \tag{4.36}$$

In both cases, as in the SBP case, the parameter α controls the upwind nature of the numerical flux. For stability $\alpha \geq 0$ with the central flux resulting if $\alpha = 0$ and the fully upwind flux if $\alpha = 1$. Note that if $\boldsymbol{p}^-$ is subtracted from $\boldsymbol{p}^*$ these are identical to the penalty terms previously defined for SBP boundaries (3.25) and conforming interfaces (3.29)–(3.30).

LEMMA 4.4. *If edge K of a DG element is an exterior boundary with penalty terms of the form (4.34) then the energy dissipation rate for the edge is*

$$\mathcal{D}_K = -\frac{\alpha}{Z}\left(\boldsymbol{p}_K^-\right)^T \boldsymbol{\Omega}_{bc} \boldsymbol{S}_{JK} \boldsymbol{p}_K^-, \tag{4.37}$$

which is negative semidefinite if $\alpha \geq 0$.

Proof. Follows directly by using (4.34) in (4.32). ∎

LEMMA 4.5. *Consider a single interface between two DG elements with penalty terms of the form (4.35)–(4.36). Let $\mathcal{D}^-$ and $\mathcal{D}^+$ be the energy dissipation rate along each side of the interface, then*

$$\begin{aligned} \mathcal{D}^- + \mathcal{D}^+ = & -\alpha \frac{Z}{2}\left(\boldsymbol{v}^- + \boldsymbol{v}^+\right)^T \boldsymbol{\Omega}_{bc} \boldsymbol{S}_J \left(\boldsymbol{v}^- + \boldsymbol{v}^+\right) \\ & - \alpha \frac{1}{2Z}\left(\boldsymbol{p}^- - \boldsymbol{p}^+\right)^T \boldsymbol{\Omega}_{bc} \boldsymbol{S}_J \left(\boldsymbol{p}^- - \boldsymbol{p}^+\right), \end{aligned} \tag{4.38}$$

is negative semidefinite for $\alpha \geq 0$.

Proof. Equation (4.38) follows directly by adding $\mathcal{D}^+ + \mathcal{D}^+$ and using the definition of the numerical flux (4.35)–(4.42). The negative semidefiniteness of (4.38) follows from the fact that $\boldsymbol{\Omega}_{bc}$ and $\boldsymbol{S}_{JK}$ are diagonal, positive definite matrices. ∎

4.2.2. SBP-to-DG interface. We now consider the case of an edge corresponding to an interface with an SBP block. Note that in general the edge of an SBP block will be connected to many DG elements as shown in Fig. 1.1 and thus a similar procedure will be required as was used in § 4.1 for connecting many SBP blocks across one interface. For simplicity, we assume that the DG element only connects to a single SBP block and that the coupling occurs along the east face of the SBP block (as shown in the right panel of Fig. 1.1). We index the glue grid using the SBP coordinate transform, so $\eta = \xi_2$ in Fig. 1.1 where ξ_2 is the second metric coordinate of the SBP block. Let the DG element intersect the glue grid over the interval $[\eta_1, \eta_2]$. Since surface Jacobian for the DG element is defined for the element's reference space on the boundary of length γ, we have to scale the surface Jacobian before projecting to the glue space as was done in the many-to-one SBP case of § 4.1. Thus we define scaled and projected DG solution as

$$\bar{\boldsymbol{v}}^- = \boldsymbol{P}^-_{f2g}\left(\frac{\boldsymbol{S}^-_J}{\Delta}\right)^{1/2} \boldsymbol{v}^-, \tag{4.39}$$

$$\bar{\boldsymbol{p}}^- = \boldsymbol{P}^-_{f2g}\left(\frac{\boldsymbol{S}^-_J}{\Delta}\right)^{1/2} \boldsymbol{p}^-, \tag{4.40}$$

where $\Delta^- = (\eta_2 - \eta_1)/\gamma$ and $\boldsymbol{P}^-_{f2g}$ is the projection from the DG element edge to the portion of the glue grid it overlaps with. Similarly we define the projection back from the glue to the DG element edge as $\boldsymbol{P}^-_{g2f}$.

If we use polynomial basis functions of order q for DG and set the glue grid space to a higher order polynomial space, then we have that $\boldsymbol{P}^-_{g2f}\boldsymbol{P}^-_{f2g} = \boldsymbol{I}$, that is there is no projection error as there was for the SBP solution; note that the converse is not true as the glue grid is a higher order space. With this assumption, we can now define the DG numerical flux when connected to the SBP finite difference solution as

$$\boldsymbol{p}^* = \left(\frac{\boldsymbol{S}^-_J}{\Delta^-}\right)^{-1/2} \boldsymbol{P}^-_{g2f}\bar{\boldsymbol{p}}^*, \tag{4.41}$$

$$\boldsymbol{v}^* - \boldsymbol{v}^- = \left(\frac{\boldsymbol{S}^-_J}{\Delta^-}\right)^{-1/2} \boldsymbol{P}^-_{g2f}\left(\bar{\boldsymbol{v}}^* - \bar{\boldsymbol{v}}^-\right), \tag{4.42}$$

with $\bar{\boldsymbol{p}}^*$ and $\bar{\boldsymbol{v}}^* - \bar{\boldsymbol{v}}^-$ defined as in (3.29)–(3.30) using the values $\bar{\boldsymbol{p}}^\pm$ and $\bar{\boldsymbol{v}}^\pm$ for $\boldsymbol{p}^\pm$ and $\boldsymbol{v}^\pm$, respectively. As in the conforming case, comparing these numerical flux expressions with the SBP penalty terms (4.2)–(4.3) we see that they are identical since there is no projection error going to the glue and back for the DG solution, in particular since

$$\boldsymbol{P}^-_{g2f}\boldsymbol{P}^-_{f2g}\boldsymbol{p}^- = \boldsymbol{P}^-_{g2f}\bar{\boldsymbol{p}}^- = \boldsymbol{p}^-. \tag{4.43}$$

Additionally, these numerical fluxes are the same as those in (4.35)–(4.36) since in the case of connecting two DG elements the projection operators are identity operations, i.e., $\boldsymbol{P}^-_{f2g} = \boldsymbol{P}^-_{g2f} = \boldsymbol{I}$, and the surface Jacobians are the same for both sides of the

interface. With the assumption that

$$\Delta^{-}\left(\boldsymbol{P}^{-}_{g2f}\right)^{T}\boldsymbol{\Omega}^{-}_{bc} = \boldsymbol{M}^{-(k)}\boldsymbol{P}^{-}_{f2g}. \tag{4.44}$$

the stability of the SBP-DG coupling is characterized by the following corollary to Theorem 4.2.

COROLLARY 4.6. *Consider a single, non-conforming interface between an SBP finite difference method and a DG method. If the SBP finite difference method has interface penalty terms of the form (4.15)–(4.16) and the DG method has numerical fluxes of the form (4.41)–(4.42), then the coupling interface satisfies the dissipation rates of Theorem 4.2.*

Proof. To prove that this corollary is true we will show that a single DG cell, indexed by $^{(k)}$, satisfies (4.21). The single edge dissipation rate for a DG cell comes from substituting (4.41)–(4.42) into (4.32) and using

$$\begin{aligned}\mathcal{D}^{-(k)} =& -\left(\boldsymbol{v}^{-(k)}\right)^{T}\boldsymbol{\Omega}_{bc}\left(\Delta^{-(k)}\boldsymbol{S}^{-(k)}_{J}\right)^{1/2}\boldsymbol{P}^{-(k)}_{g2f}\bar{\boldsymbol{p}}^{*(k)}\\ &-\left(\boldsymbol{p}^{-(k)}\right)^{T}\boldsymbol{\Omega}_{bc}\left(\Delta^{-(k)}\boldsymbol{S}^{-(k)}_{J}\right)^{1/2}\boldsymbol{P}^{-(k)}_{g2f}\left(\bar{\boldsymbol{v}}^{*(k)}-\bar{\boldsymbol{v}}^{-(k)}\right).\end{aligned} \tag{4.45}$$

Using (4.44) this can be simplified to (4.21). Thus, the rest of the proof for the dissipation rates follow the same procedure as in the proof of Theorem 4.2. ∎

5. Numerical Results. Here we confirm the above theoretical stability results as well as explore the accuracy of the coupling technique.[1] A method-of-lines approach is use to discretize the acoustic wave equation where the spatial schemes is as described in this paper and an explicit 4th order Runge–Kutta method is used for the temporal discretization. The test problem is the discretization of (3.4)–(3.5) on the domain $\Omega = [-1,1]\times[-1,1]$ with $\rho = \lambda = 1$. Zero pressure, i.e., free surface, boundary conditions are used on all boundaries. We use the initial condition

$$p(x_1,x_2,0) = \cos(k_1x_1)\cos(k_1x_2) + \sin(k_2x_1)\sin(k_2x_2), \tag{5.1}$$

$$v_i(x_1,x_2,0) = 0, \qquad i = 1,2, \tag{5.2}$$

where $k_1 = \pi/2$ and $k_2 = \pi$. With this the exact solution is

$$p(x_1,x_2,t) = \cos(\omega_1 t)\cos(k_1x_1)\cos(k_1x_2) + \cos(\omega_2 t)\sin(k_2x_1)\sin(k_2x_2), \tag{5.3}$$

$$v_1(x_1,x_2,t) = \frac{k_1}{\omega_1}\sin(\omega_1 t)\sin(k_1x_1)\cos(k_1x_2) - \frac{k_2}{\omega_2}\sin(\omega_2 t)\cos(k_2x_1)\sin(k_2x_2), \tag{5.4}$$

$$v_2(x_1,x_2,t) = \frac{k_1}{\omega_1}\sin(\omega_1 t)\cos(k_1x_1)\sin(k_1x_2) - \frac{k_2}{\omega_2}\sin(\omega_2 t)\sin(k_2x_1)\cos(k_2x_2), \tag{5.5}$$

where $\omega_j = k_j\sqrt{2}$ for $j = 1,2$.

[1] MATLAB code for constructing the interpolation operators used in this section are available at `https://github.com/bfam/sbp_projection_operators`. The simulation code used to produce the results is available at `https://github.com/bfam/sbp_projection_2d`. For DG, when coupling with SBP-SAT, we use the code from Hesthaven and Warburton [5] available at `https://github.com/tcew/nodal-dg`.

N	$q=2$ error (rate)	$q=3$ error (rate)	$q=4$ error (rate)	$q=5$ error (rate)
		conforming meshes (no projection)		
64	4.3×10^{-4}	2.1×10^{-4}	4.6×10^{-5}	3.7×10^{-5}
128	5.2×10^{-5} (3.0)	1.4×10^{-5} (3.9)	1.4×10^{-6} (5.0)	5.7×10^{-7} (6.0)
256	6.5×10^{-6} (3.0)	9.3×10^{-7} (4.0)	4.8×10^{-8} (4.9)	7.7×10^{-9} (6.2)
512	8.0×10^{-7} (3.0)	5.9×10^{-8} (4.0)	1.9×10^{-9} (4.6)	1.1×10^{-10} (6.1)
1024	1.0×10^{-7} (3.0)	3.7×10^{-9} (4.0)	7.6×10^{-11} (4.6)	1.7×10^{-12} (6.0)
		nested meshes		
64	5.5×10^{-4}	1.4×10^{-4}	8.3×10^{-5}	6.5×10^{-5}
128	7.8×10^{-5} (2.8)	7.9×10^{-6} (4.2)	6.6×10^{-6} (3.7)	1.3×10^{-6} (5.7)
256	1.1×10^{-5} (2.8)	4.8×10^{-7} (4.0)	3.1×10^{-7} (4.4)	1.5×10^{-8} (6.4)
512	1.6×10^{-6} (2.8)	3.2×10^{-8} (3.9)	1.3×10^{-8} (4.5)	1.2×10^{-10} (7.0)
1024	2.2×10^{-7} (2.8)	2.3×10^{-9} (3.8)	6.4×10^{-10} (4.4)	1.5×10^{-12} (6.3)
		unnested meshes		
64	5.0×10^{-4}	1.4×10^{-4}	8.9×10^{-5}	6.2×10^{-5}
128	7.1×10^{-5} (2.8)	8.0×10^{-6} (4.2)	6.7×10^{-6} (3.7)	1.2×10^{-6} (5.6)
256	1.0×10^{-5} (2.8)	4.8×10^{-7} (4.1)	3.1×10^{-7} (4.4)	1.4×10^{-8} (6.4)
512	1.5×10^{-6} (2.8)	3.2×10^{-8} (3.9)	1.3×10^{-8} (4.5)	1.1×10^{-10} (7.0)
1024	2.1×10^{-7} (2.8)	2.3×10^{-9} (3.8)	6.3×10^{-10} (4.4)	1.5×10^{-12} (6.3)

TABLE 5.1

Table of the error and estimated convergence rates for the coupling of two SBP blocks (see simulations shown in Fig. 5.1).

5.1. One-to-One SBP Coupling. We first test the coupling of two SBP blocks as shown in the left panel of Fig. 1.1. The coordinate transforms for the two blocks is

$$x_1^{(L)}(\xi_1,\xi_2) = \left(\frac{1+\xi_1}{10}\right)\sin\left(\pi\left(\xi_2+1\right)\right) - \left(\frac{1-\xi_1}{2}\right), \qquad x_2^{(L)}(\xi_1,\xi_2) = \xi_2, \tag{5.6}$$

$$x_1^{(R)}(\xi_1,\xi_2) = \left(\frac{1-\xi_1}{10}\right)\sin\left(\pi\left(\xi_2+1\right)\right) + \left(\frac{1+\xi_1}{2}\right), \qquad x_2^{(R)}(\xi_1,\xi_2) = \xi_2, \tag{5.7}$$

where the superscript (L) and (R) corresponds to the left and right block, respectively. Notice that the coordinate transform along the interface is conforming $x_1^{(L)}(1,\xi_2) = x_1^{(R)}(-1,\xi_2) = \frac{1}{5}\sin\left(\pi\left(\xi_2+1\right)\right)$ and $x_2^{(L)}(1,\xi_2) = x_2^{(R)}(-1,\xi_2) = \xi_2$.

We discretize the left block with using an $(N/2+1)\times(N+1)$ grid of where $N=2^k$ with $k=6,7,8,9,10$. For the right block we use an $(M/2+1)\times(M+1)$ grid of points where M is chosen so the interface if conforming $(M=N)$, nested $(M=2N)$, or unnested $(M=2N+1)$. In the conforming case no projection operator is used, i.e., this is the traditional SBP-SAT coupling. We run the simulation using SBP orders $q=2,3,4,5$, where here q refers to the boundary order, i.e., the interior finite difference method is of order $p_i=2q$, the boundary finite difference order is $p_b=q$, and the expected rate of convergence for conforming multiblock SBP is $q+1$. Note that in this work we will exclusively consider the diagonal norm SBP operators. The final time of the simulations is $t=1$. For the penalties we use $\alpha=1$, thus the interface and boundaries are fully upwinded.

In Fig. 5.1 and Table 5.1 we report the error for each of the three cases. We

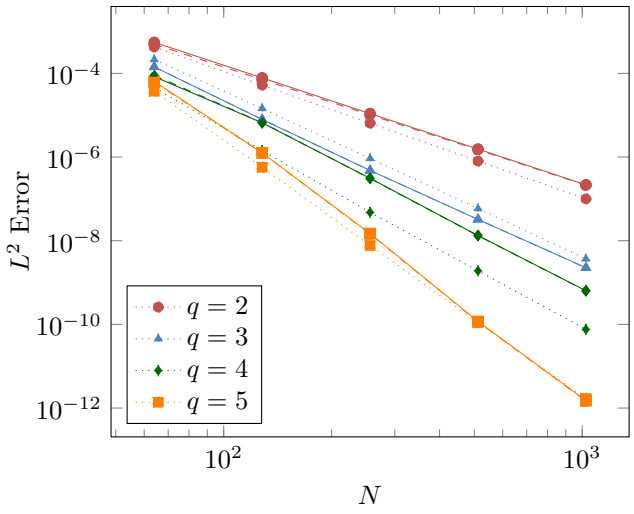

FIG. 5.1. *Log-log plot of N versus the L^2 error (measured with the energy norm) for the coupling of two SBP blocks (see Fig. 1.1). Three different interface types are shown: a conforming interface (dotted line), a nested interface with a two-to-one refinement ratio (solid line), and an unnested interface with $N_{fine} = 2N_{coarse} + 1$ (dashed line). In the figure axis N refers to the coarse block, i.e., $N_{coarse} = N$.*

measure the error using the L^2 norm

$$\varepsilon^2 = \sum_{\{\Omega_e\}} \left(\frac{\rho}{2} \boldsymbol{\Delta v}_1^T \boldsymbol{J} \bar{\boldsymbol{H}} \boldsymbol{\Delta v}_1 + \frac{\rho}{2} \boldsymbol{\Delta v}_2^T \boldsymbol{J} \bar{\boldsymbol{H}} \boldsymbol{\Delta v}_2 + \frac{1}{2\lambda} \boldsymbol{\Delta p}^T \boldsymbol{J} \bar{\boldsymbol{H}} \boldsymbol{\Delta p} \right), \tag{5.8}$$

where $\boldsymbol{\Delta v}_1$, $\boldsymbol{\Delta v}_2$, and $\boldsymbol{\Delta p}$ are the difference between the discrete solution and exact solution at all the grid points; note that this is the same norm used in the stability analysis (3.20). In all the cases the error is decreasing with mesh refinement. For both of the cases where the projection operators are used, the overall error level is comparable and slightly higher than the conforming (no projection) case. Also given in Table 5.1 are estimates of the convergence rate between two successive resolutions measured using

$$\frac{\log(\varepsilon_f) - \log(\varepsilon_c)}{\log(N_c) - \log(N_f)}, \tag{5.9}$$

where here subscript f refers to the finer solution and c the coarser solution. As the table shows, the coupling does show higher-order convergence though it is interesting to note that when the projection are used the rates are more sporadic than the conforming case.

5.2. Two-to-One SBP Coupling. We now test the use of the projection operators to couple multiple SBP blocks along a single interface. That is we have a single block on the left side of the interface and two blocks on the right side of the interface as in the center panel of Fig. 1.1. The block on the left side of the interface has the coordinate transform (5.6), where as the top and bottom right blocks have

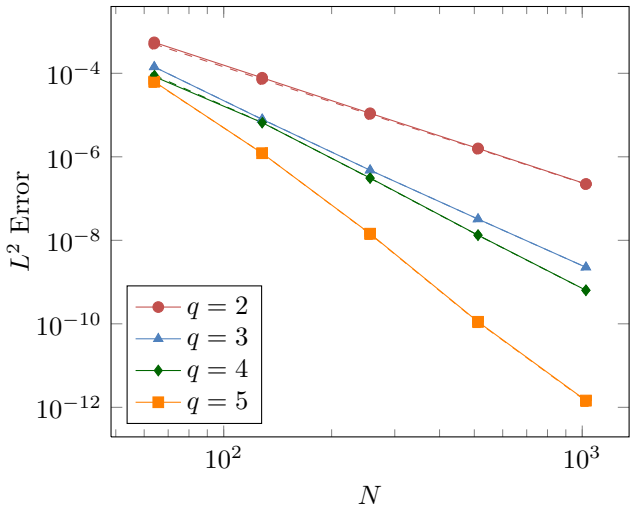

FIG. 5.2. *Log-log plot of N versus the L^2 error (measured with the energy norm) for the coupling of three SBP blocks (see Fig. 1.1). Two different interface types are shown: a nested interface with a two-to-one refinement ratio (solid line) and an unnested interface with $N_{fine} = 2N_{coarse} + 1$ (dashed line). Also shown for reference is the two block conforming interface test (dotted line) from Fig. 5.1. In the figure axis N refers to the coarse block, i.e., $N_{coarse} = N$.*

the coordinates transforms:

$$x_1^{(RT)}(\xi_1, \xi_2) = x_1^{(R)}\left(\frac{\xi_1 + 1}{2}, \xi_2\right), \qquad x_2^{(RT)}(\xi_1, \xi_2) = x_2^{(R)}\left(\frac{\xi_1 + 1}{2}, \xi_2\right), \tag{5.10}$$

$$x_1^{(RB)}(\xi_1, \xi_2) = x_1^{(R)}\left(\frac{\xi_1 - 1}{2}, \xi_2\right), \qquad x_2^{(RB)}(\xi_1, \xi_2) = x_2^{(R)}\left(\frac{\xi_1 - 1}{2}, \xi_2\right), \tag{5.11}$$

where superscript (RT) and (RB) refer to the right-top and right-bottom blocks, respectively, and $x_{1,2}^{(R)}$ are defined by (5.7). Notice that once again all the interfaces are conforming at the continuous level.

We discretize the left block with an $(N/2 + 1) \times (N + 1)$ grid of points, with $N = 2^k$ for $k = 6, 7, 8, 9, 10$. The right two blocks are discretized using grids of size $(N+1)\times(M+1)$ where is chosen so that the interface is nested with a $1 : 2$ refinement ratio $(M = N)$ or fully unnested $(M = N + 1)$; in both cases the interface between the two right blocks is conforming. As before we let the final time be $t = 1$ and use $\alpha = 1$ in the penalty terms.

In Fig. 5.2 and Table 5.2 we report the error (as measured by (5.8)) for SBP orders $q = 2, 3, 4, 5$, where as before q refers to the boundary order of the SBP method. In Table 5.2 we also report the convergence rate as calculated using (5.9). As these results show, the method maintains the high-order accuracy of the SBP finite difference when non-conforming block interfaces are used. As in the case of the one-to-one coupling, the rates are more sporadic than the conforming one-to-one coupling case.

5.3. SBP-DG Coupling. Here we consider the coupling between SBP finite difference methods and DG finite elements methods as discussed in § 4.2. We specifically use the curvilinear nodal DG method on triangles as described in Hesthaven and Warburton [5]. The configuration is as shown in the right panel of Fig. 1.1 with an curvilinear SBP block on the left side of the coupling interface and an unstructured,

N	$q=2$ error (rate)	$q=3$ error (rate)	$q=4$ error (rate)	$q=5$ error (rate)
		nested meshes		
64	5.5×10^{-4}	1.4×10^{-4}	8.4×10^{-5}	6.3×10^{-5}
128	7.8×10^{-5} (2.8)	7.9×10^{-6} (4.2)	6.6×10^{-6} (3.7)	1.2×10^{-6} (5.7)
256	1.1×10^{-5} (2.8)	4.8×10^{-7} (4.0)	3.1×10^{-7} (4.4)	1.4×10^{-8} (6.4)
512	1.6×10^{-6} (2.8)	3.2×10^{-8} (3.9)	1.3×10^{-8} (4.5)	1.1×10^{-10} (7.0)
1024	2.2×10^{-7} (2.8)	2.3×10^{-9} (3.8)	6.4×10^{-10} (4.4)	1.5×10^{-12} (6.3)
		unnested meshes		
64	5.0×10^{-4}	1.4×10^{-4}	8.9×10^{-5}	6.2×10^{-5}
128	7.1×10^{-5} (2.8)	7.9×10^{-6} (4.2)	6.7×10^{-6} (3.8)	1.2×10^{-6} (5.7)
256	1.0×10^{-5} (2.8)	4.8×10^{-7} (4.1)	3.1×10^{-7} (4.4)	1.4×10^{-8} (6.4)
512	1.5×10^{-6} (2.8)	3.2×10^{-8} (3.9)	1.3×10^{-8} (4.5)	1.1×10^{-10} (7.0)
1024	2.2×10^{-7} (2.8)	2.3×10^{-9} (3.8)	6.3×10^{-10} (4.4)	1.4×10^{-12} (6.3)

TABLE 5.2

Table of the error and estimated convergence rates for the coupling of three SBP blocks (see simulations shown in Fig. 5.2).

N	$q=2$ error (rate)	$q=3$ error (rate)	$q=4$ error (rate)	$q=5$ error (rate)
64	5.3×10^{-4}	1.7×10^{-4}	6.7×10^{-5}	8.1×10^{-5}
128	8.6×10^{-5} (2.6)	9.0×10^{-6} (4.3)	5.7×10^{-6} (3.6)	1.3×10^{-6} (5.9)
256	2.0×10^{-5} (2.1)	4.9×10^{-7} (4.2)	3.2×10^{-7} (4.1)	1.3×10^{-8} (6.7)
512	5.8×10^{-6} (1.8)	3.5×10^{-8} (3.8)	1.5×10^{-8} (4.4)	1.1×10^{-10} (6.8)
1024	1.4×10^{-6} (2.0)	4.7×10^{-9} (2.9)	6.3×10^{-10} (4.6)	1.7×10^{-12} (6.1)

TABLE 5.3

SBP-DG. Where q is the SBP boundary order and the DG polynomial order

curvilinear DG mesh on the right side of the coupling interface. The SBP block is transformed according to (5.6). For the DG mesh, the element edges along the curved interface are curved by moving the interpolation nodes to the interface and the interior interpolation points are then moved using transfinite blending; elements that are not on the interface remain straight-sided. Refinement for the unstructured mesh is performed in a hierarchical fashion with each triangle split into four nested triangles.

As before, we run the SBP mesh with SBP orders $q = 2, 3, 4, 5$ and use a polynomial order for the DG elements of q. Similarly, we discretize the left SBP block with an $(N/2+1) \times (N+1)$ grid of points, with $N = 2^k$ for $k = 6, 7, 8, 9, 10$. The initial DG mesh is chosen so that the number of unique degrees of freedom along the interface roughly matches the number of finite difference grid points. That is, we choose a base mesh for each order that has $\lceil 2^6/(q+1) \rceil$ edges along the coupling interface which is then refined in a hierarchical manner. The final time for the simulations is $t = 1$ and $\alpha = 1$ is used in both the SBP penalties and DG numerical flux terms.

Shown in Fig. 5.3 and Table 5.3 are the error and convergence results for this test problem. As can be seen the method is converging at high-order accuracy. As in the purely SBP to SBP coupling the convergence rates are rather sporadic.

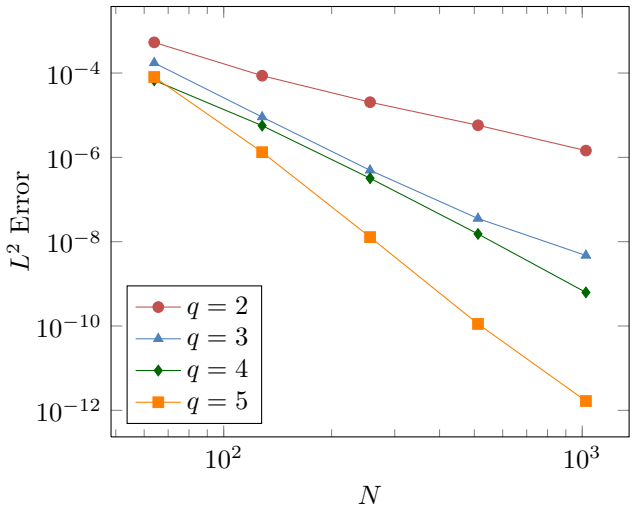

FIG. 5.3. *Log-log plot of N versus the L^2 error (measured with the energy norm) for the coupling of SBP and DG (see Fig. 1.1). We plot different lines for runs of different orders where q is both the SBP boundary order and the DG polynomial order.*

Eigenvalue Spectrum. Here we confirm the stability results by looking at the eigenvalue spectrum of the SBP-DG coupling. To do this we write the fully coupled system as a linear equation

$$\frac{d\boldsymbol{u}}{dt} = \boldsymbol{A}\boldsymbol{u}, \tag{5.12}$$

and then numerically compute the eigenvalues of $\boldsymbol{A}$. The energy stability analysis implies that all the eigenvalues should have a non-positive real part. Furthermore, when the penalty/numerical flux parameter is chosen to be $\alpha = 0$ then the eigenvalues should be purely imaginary. To confirm this, in Fig. 5.4 we show the eigenvalue value spectrum for the of the coupling of the SBP operator with $q = 5$ with the DG using polynomial order 5; the mesh configuration used is the first resolution for this coupling from § 5.3. The maximum real part of the eigenvalue spectrum is 8.16×10^{-13} for the upwind penalty ($\alpha = 1$) and the maximum magnitude real part of the eigenvalue spectrum is 2.67×10^{-13} for the central penalty ($\alpha = 0$), thus confirming the theoretical stability results.

6. Conclusions. In this paper we have presented a new approach to coupling high-order finite difference methods across non-conforming grid interfaces as well as with DG methods. The core idea behind the proposed methodology is the construction of a projection operator that moves the grid solution from the finite difference points to a finite dimension subspace of the Hilbert space $L^2(\Gamma)$. The value of this is that once the solution is in this subspace it can be projected using L^2 integral projections to other finite dimensions subspaces of the Hilbert space $L^2(\Gamma)$ and then projected back to the finite difference grid. Since the projection operators are consistent with the SBP H-matrix the fully coupled method is provable stable through the use of weak enforcement of boundary conditions. In addition to enforcing the boundary conditions weakly, it was necessary to account for the error in the projection operator (namely, the fact that the finite difference grid space and the finite dimensional subspace are not hierarchical spaces).

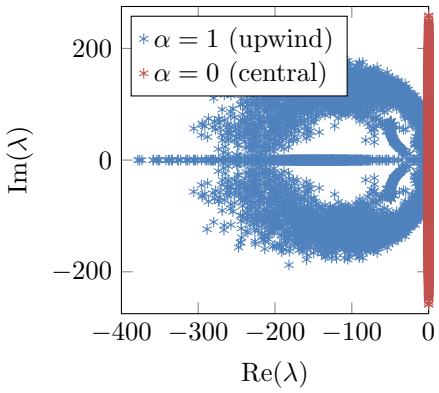

FIG. 5.4. *Eigenvalues of the coupled SBP-DG spatial discretization operator for the coupling of the SBP operator with $q = 5$ with the DG using polynomial order 5; the mesh configuration used is the first resolution for this coupling from § 5.3. We show results for both upwind $\alpha = 1$ and central $\alpha = 0$ penalties.*

In this work we chose a subspace of piecewise polynomial functions which conformed with the finite difference points. The order of the polynomials used matched the structure of the SBP finite difference method. That is we required that the resulting projection operators be exact in the interior for polynomials of order $q_i - 1$ and near the boundary for polynomials of order $q_b - 1$ with q_i and q_b being the interior and boundary accuracy of the SBP finite difference method.

This choice of piecewise polynomial functions as the intermediate space is tangential to the stability results, and any other finite dimensions subspace of the Hilbert space $L^2(\Gamma)$ could have been chosen. The reason for this is that the projection operators we have constructed to move to polynomials can be used as an intermediate step in moving to any other finite dimensional subspace. That said, the value of another subspace (including a different set of piecewise polynomials) would lie in the ability to enforce different accuracy conditions which may be used to improve the error.

In addition to proposing a new class of SBP-compatible projection operators, we also showed how these projection operators can be used to account for differences in the discrete geometry seen by different blocks on either side of an interface. Namely, we showed that the stability of the numerical method could be preserved if the geometry terms were projected through with the grid values. In all the work we have done here we assume that at the continuous level the coordinate transforms are conforming. The stability results carryover to the case of dissimilar continuous coordinate transforms, but the approach outlined in this paper reduces the results to first order accurate. It is possible that high-order accuracy can still be preserved in the case of dissimilar coordinate transforms if high-order accurate projection operators were constructed between the coordinate transforms themselves, but this was not explored in this work.

Appendix A. Proof of Lemma 3.2. Taking the time derivative of the energy norm (3.20) and substituting in discretization (3.12)–(3.13) gives the energy dissipa-

tion rate

$$\begin{aligned}\frac{dE}{dt} = & -\boldsymbol{v}_1^T\boldsymbol{Q}_1\boldsymbol{J}\frac{\boldsymbol{\partial\xi_1}}{\boldsymbol{\partial x_1}}\boldsymbol{p} - \boldsymbol{v}_1^T\boldsymbol{Q}_2\boldsymbol{J}\frac{\boldsymbol{\partial\xi_2}}{\boldsymbol{\partial x_1}}\boldsymbol{p} - \boldsymbol{v}_2^T\boldsymbol{Q}_1\boldsymbol{J}\frac{\boldsymbol{\partial\xi_1}}{\boldsymbol{\partial x_2}}\boldsymbol{p} - \boldsymbol{v}_2^T\boldsymbol{Q}_2\boldsymbol{J}\frac{\boldsymbol{\partial\xi_2}}{\boldsymbol{\partial x_2}}\boldsymbol{p} \\ & -\boldsymbol{p}^T\boldsymbol{J}\frac{\boldsymbol{\partial\xi_1}}{\boldsymbol{\partial x_1}}\boldsymbol{Q}_1\boldsymbol{v_1} - \boldsymbol{p}^T\boldsymbol{J}\frac{\boldsymbol{\partial\xi_2}}{\boldsymbol{\partial x_1}}\boldsymbol{Q}_2\boldsymbol{v_1} - \boldsymbol{p}^T\boldsymbol{J}\frac{\boldsymbol{\partial\xi_1}}{\boldsymbol{\partial x_2}}\boldsymbol{Q}_1\boldsymbol{v_2} - \boldsymbol{p}^T\boldsymbol{J}\frac{\boldsymbol{\partial\xi_2}}{\boldsymbol{\partial x_2}}\boldsymbol{Q}_2\boldsymbol{v_2} \\ & -\boldsymbol{v}_1^T\boldsymbol{\mathcal{F}}_{v_1} - \boldsymbol{v}_2^T\boldsymbol{\mathcal{F}}_{v_2} - \boldsymbol{p}^T\boldsymbol{\mathcal{F}}_p,\end{aligned} \tag{A.1}$$

where $\boldsymbol{Q}_1 = \boldsymbol{Q}_{N_1} \otimes \boldsymbol{H}_{N_2}$ and $\boldsymbol{Q}_2 = \boldsymbol{H}_{N_1} \otimes \boldsymbol{Q}_{N_2}$. Using the SBP property $\boldsymbol{Q} + \boldsymbol{Q}^T = \boldsymbol{B} = \text{diag}[-1, 0, \ldots, 0, 1]$ and the fact that $\boldsymbol{J}$ and $\frac{\boldsymbol{\partial\xi_j}}{\boldsymbol{\partial x_i}}$ are diagonal matrices (and thus commute), the volume terms can be transformed to boundary terms:

$$\begin{aligned}\frac{dE}{dt} = & -\boldsymbol{v}_1^T\boldsymbol{B}_1\boldsymbol{J}\frac{\boldsymbol{\partial\xi_1}}{\boldsymbol{\partial x_1}}\boldsymbol{p} - \boldsymbol{v}_1^T\boldsymbol{B}_2\boldsymbol{J}\frac{\boldsymbol{\partial\xi_2}}{\boldsymbol{\partial x_1}}\boldsymbol{p} - \boldsymbol{v}_2^T\boldsymbol{B}_1\boldsymbol{J}\frac{\boldsymbol{\partial\xi_1}}{\boldsymbol{\partial x_2}}\boldsymbol{p} - \boldsymbol{v}_2^T\boldsymbol{B}_2\boldsymbol{J}\frac{\boldsymbol{\partial\xi_2}}{\boldsymbol{\partial x_2}}\boldsymbol{p} \\ & -\boldsymbol{v}_1^T\boldsymbol{\mathcal{F}}_{v_1} - \boldsymbol{v}_2^T\boldsymbol{\mathcal{F}}_{v_2} - \boldsymbol{p}^T\boldsymbol{\mathcal{F}}_p \\ = & -\boldsymbol{v}_W^T\boldsymbol{H}_2\boldsymbol{S}_{JW}\boldsymbol{p}_W - \boldsymbol{v}_E^T\boldsymbol{H}_2\boldsymbol{S}_{JE}\boldsymbol{p}_E - \boldsymbol{v}_S^T\boldsymbol{H}_1\boldsymbol{S}_{JS}\boldsymbol{p}_S - \boldsymbol{v}_N^T\boldsymbol{H}_1\boldsymbol{S}_{JN}\boldsymbol{p}_N \\ & -\boldsymbol{v}_1^T\boldsymbol{\mathcal{F}}_{v_1} - \boldsymbol{v}_2^T\boldsymbol{\mathcal{F}}_{v_2} - \boldsymbol{p}^T\boldsymbol{\mathcal{F}}_p,\end{aligned} \tag{A.2}$$

where $\boldsymbol{B}_1 = \boldsymbol{B}_{N_1} \otimes \boldsymbol{H}_{N_2}$ and $\boldsymbol{B}_2 = \boldsymbol{H}_{N_1} \otimes \boldsymbol{B}_{N_2}$. Note that here we have also used the fact that along the block boundaries

$$J\frac{\partial\xi_i}{\partial x_2} = \pm n_1 S_J, \qquad\qquad J\frac{\partial\xi_i}{\partial x_1} = \pm n_2 S_J, \tag{A.3}$$

with the positive sign being taken on the "north" ($\xi_2 = 1$) and "east" ($\xi_1 = 1$) boundaries and the negative sign being taken on the on the "south" ($\xi_2 = -1$) and "east" ($\xi_1 = -1$) boundaries; see (3.8). Recall also that $\boldsymbol{v}_W$, $\boldsymbol{v}_E$, $\boldsymbol{v}_N$, and $\boldsymbol{v}_S$ are the normal components of the velocity along the west, east, north, and south edges; see (3.19). Using the definition of the penalty terms (3.16)–(3.17) along with form (3.19), allows us to rewrite the penalty terms in (A.2) as

$$\begin{aligned}\boldsymbol{v}_1^T\boldsymbol{\mathcal{F}}_{v_1} + \boldsymbol{v}_2^T\boldsymbol{\mathcal{F}}_{v_2} = & \boldsymbol{v}_W^T\boldsymbol{H}_2\boldsymbol{S}_{JW}\left(\boldsymbol{p}_W^* - \boldsymbol{p}_W\right) + \boldsymbol{v}_E^T\boldsymbol{H}_2\boldsymbol{S}_{JE}\left(\boldsymbol{p}_E^* - \boldsymbol{p}_E\right) \\ & + \boldsymbol{v}_S^T\boldsymbol{H}_1\boldsymbol{S}_{JS}\left(\boldsymbol{p}_S^* - \boldsymbol{p}_S\right) + \boldsymbol{v}_N^T\boldsymbol{H}_1\boldsymbol{S}_{JN}\left(\boldsymbol{p}_N^* - \boldsymbol{p}_N\right)\end{aligned} \tag{A.4}$$

$$\begin{aligned}\boldsymbol{p}^T\boldsymbol{\mathcal{F}}_p = & \boldsymbol{p}_W^T\boldsymbol{H}_2\boldsymbol{S}_{JW}\left(\boldsymbol{v}_W^* - \boldsymbol{v}_W\right) + \boldsymbol{p}_E^T\boldsymbol{H}_2\boldsymbol{S}_{JE}\left(\boldsymbol{v}_E^* - \boldsymbol{v}_E\right) \\ & + \boldsymbol{p}_S^T\boldsymbol{H}_1\boldsymbol{S}_{JS}\left(\boldsymbol{v}_S^* - \boldsymbol{v}_S\right) + \boldsymbol{p}_N^T\boldsymbol{H}_1\boldsymbol{S}_{JN}\left(\boldsymbol{v}_N^* - \boldsymbol{v}_N\right).\end{aligned} \tag{A.5}$$

These penalty terms can then be used in (A.2) to write

$$\begin{aligned}\frac{dE}{dt} = & -\boldsymbol{v}_W^T\boldsymbol{H}_2\boldsymbol{S}_{JW}\boldsymbol{p}_W^* + \boldsymbol{v}_W^T\boldsymbol{H}_2\boldsymbol{S}_{JW}\boldsymbol{p}_W - \left(\boldsymbol{v}_W^*\right)^T\boldsymbol{H}_2\boldsymbol{S}_{JW}\boldsymbol{p}_W \\ & -\boldsymbol{v}_E^T\boldsymbol{H}_2\boldsymbol{S}_{JE}\boldsymbol{p}_E^* + \boldsymbol{v}_E^T\boldsymbol{H}_2\boldsymbol{S}_{JE}\boldsymbol{p}_E - \left(\boldsymbol{v}_E^*\right)^T\boldsymbol{H}_2\boldsymbol{S}_{JE}\boldsymbol{p}_E \\ & -\boldsymbol{v}_S^T\boldsymbol{H}_1\boldsymbol{S}_{JS}\boldsymbol{p}_S^* + \boldsymbol{v}_S^T\boldsymbol{H}_1\boldsymbol{S}_{JS}\boldsymbol{p}_S - \left(\boldsymbol{v}_S^*\right)^T\boldsymbol{H}_1\boldsymbol{S}_{JS}\boldsymbol{p}_S \\ & -\boldsymbol{v}_N^T\boldsymbol{H}_1\boldsymbol{S}_{JN}\boldsymbol{p}_N^* + \boldsymbol{v}_N^T\boldsymbol{H}_1\boldsymbol{S}_{JN}\boldsymbol{p}_N - \left(\boldsymbol{v}_N^*\right)^T\boldsymbol{H}_1\boldsymbol{S}_{JN}\boldsymbol{p}_N,\end{aligned} \tag{A.6}$$

which is (3.23) with the substitution of (3.24).

Appendix B. Projection Operators. Here we discuss the construction of projection operators that satisfy Definition 2.2 and the accuracy conditions (2.5). We

will first consider the construction of the operators in the interior of the domain and then the operator near the boundary. We only consider diagonal norm SBP operators that have boundary accuracy p_b and interior accuracy $p_i = 2p_b$. Thus from (2.5) we are thus seeking projection operators $\boldsymbol{P}_{f2g}$ and $\boldsymbol{P}_{g2f}$ which exactly project interior polynomials of order $p_i - 1$ and boundary polynomials of order $p_b - 1$.

Let the glue grid be represented by the basis $\{\phi_i\}_{i=0}^n$ where $n = p_i - 1$; we use orthonormalized Legendre polynomials defined on $[-1, 1]$ in our codes. The piecewise polynomial on the kth glue grid interval, e.g., the glue grid interval $[x_k, x_{k-1}]$, is then $p_n^{(k)} = \sum_{i=0}^n \omega_i^{(k)} \phi_i^{(k)}(x)$. Note that here (and following) the superscript (k) on $\phi_i^{(k)}(x)$ is to signify that polynomial has been shifted to the interval $[x_k, x_{k+1}]$, i.e.,

$$\phi_i^{(k)}(x) = \phi_i \left(\frac{x - x_k}{x_{k+1} - x_k} - \frac{x_{k+1} - x}{x_{k+1} - x_k} \right). \tag{B.1}$$

In the interior, $m = 2l$ glue grid intervals are projected to a single grid point. Thus, if we consider grid point q_k we use intervals $k + 1 - l$ through $k + l$. It is also natural to impose symmetry conditions on the operator, which then leads to a projection of the form

$$q_k = \sum_{j=1}^{l} \sum_{i=0}^{n} \beta_{ij} \left[(-1)^i \omega_i^{(k+1-j)} + \omega_i^{(k+j)} \right], \tag{B.2}$$

where the $(-1)^i$ comes from the fact that even modes are symmetric and odd modes skew-symmetric. Here the coefficient β_{ij} is the contribution of mode i of interval $k+1-j$ and $k+j$ to the grid point value k. Note that β_{ij} are the values that we are seeking to construct.

For the $s + 1$ grid points near the boundary, e.g., grid values q_k for $k = 0, \ldots, s$, the intervals 1 through r are used to construct the projection; by the conditions for stability given in Definition 2.2 it is required that $s + 1 = r + 1 - l$. The projection at the boundary then takes the form

$$q_k = \sum_{i=0}^{n} \sum_{j=0}^{r-1} \alpha_{ij}^{(k)} \omega_i^{(j)}, \ k = 0, \ldots, s, \tag{B.3}$$

where here we allow the boundary polynomials to be of order n even though the accuracy condition at the boundary are for polynomials of degree $p_b - 1 = (n - 1)/2$. Hence in total we have $m(n+1)$ values β_{ij} and $r(s+1)(n+1)$ values $\alpha_{ij}^{(k)}$ to determine.

It is important to remember that by Definition 2.2, the structure of the projection from the finite difference grid to the glue is given by $\boldsymbol{P}_{f2g} = \boldsymbol{M}^{-1} \boldsymbol{P}_{g2f}^T \boldsymbol{H}$. Thus, once the structure of $\boldsymbol{P}_{g2f}$ is specified $\boldsymbol{P}_{f2g}$ is also fixed, i.e., no new degrees of freedom are introduced in the problem.

Equations (B.2) and (B.3) along with accuracy condition (2.5) and the projection compatibility conditions (2.4) induce a linear system of equations that must be solved. We solve these equations numerically using the MATLAB. The values of s and l needed for the SBP operators used in this work are given in Table B.1. In determining the projection coefficients β_{ij} and $\alpha_{ij}^{(k)}$ degrees of freedom remain in the problem, we use MATLAB's optimization library to minimize the distance between nearest eigenvalues of $\boldsymbol{B} = \boldsymbol{P}_{g2f} \boldsymbol{P}_{f2g}$ for a finite difference grid of size N. The motivation for this optimization is to (in some sense) minimize the projection error by using the degrees

p_i	p_b	l	$s+1$	r
2	1	1	1	1
4	2	2	4	5
6	3	3	6	8
8	4	4	9	12
10	5	5	12	16

TABLE B.1
Parameters used in construction of the SBP compatible projection operators.

of freedom to make $\boldsymbol{B}$ closer to an identity operation. We use the value of $N = 64$, which has been chosen to make the optimization problem tractable. It is important to note that the optimization is being used to fix the remaining degrees of freedom after the stability condition (2.4) is satisfied, i.e., stability does not depend on the choice of the optimization objective function and other optimization could be considered.

MATLAB routines as well as the final optimized coefficients can be found in the github repository located at `https://github.com/bfam/sbp_projection_operators`.

REFERENCES

[1] M. H. Carpenter, D. Gottlieb, and S. Abarbanel, *Time-stable boundary conditions for finite-difference schemes solving hyperbolic systems: Methodology and application to high-order compact schemes*, Journal of Computational Physics, 111 (1994), pp. 220–236.

[2] G. Chesshire and W. D. Henshaw, *Composite overlapping meshes for the solution of partial differential equations*, Journal of Computational Physics, 90 (1990), p. 164.

[3] B. Gustafsson, *The convergence rate for difference approximations to mixed initial boundary value problems*, Mathematics of Computation, 29 (1975), pp. 396–406.

[4] B. Gustafsson, H.-O. Kreiss, and J. Oliger, *Time Dependent Problems and Difference Methods*, Wiley-Interscience, New York, 1996.

[5] Jan S. Hesthaven and Tim Warburton, *Nodal Discontinuous Galerkin Methods: Algorithms, Analysis, and Applications*, vol. 54 of Texts in Applied Mathematics, Springer, 2008.

[6] J. E. Kozdon, E. M. Dunham, and J. Nordström, *Interaction of waves with frictional interfaces using summation-by-parts difference operators: Weak enforcement of nonlinear boundary conditions*, Journal of Scientific Computing, 50 (2012), pp. 341–367.

[7] ———, *Simulation of dynamic earthquake ruptures in complex geometries using high-order finite difference methods*, Journal of Scientific Computing, 55 (2013), pp. 92–124.

[8] H.-O. Kreiss and G. Scherer, *Finite element and finite difference methods for hyperbolic partial differential equations*, in Mathematical aspects of finite elements in partial differential equations; Proceedings of the Symposium, Madison, WI, 1974, pp. 195–212.

[9] ———, *On the existence of energy estimates for difference approximations for hyperbolic systems*, tech. report, Dept. of Scientific Computing, Uppsala University, 1977.

[10] K. Mattsson and M. Almquist, *A solution to the stability issues with block norm summation by parts operators*, Journal of Computational Physics, 253 (2013), pp. 418–442.

[11] K. Mattsson and M. Carpenter, *Stable and accurate interpolation operators for high-order multiblock finite difference methods*, SIAM Journal on Scientific Computing, 32 (2010), pp. 2298–2320.

[12] K. Mattsson and J. Nordström, *Summation by parts operators for finite difference approximations of second derivatives*, Journal of Computational Physics, 199 (2004), pp. 503–540.

[13] A. Nissen, K. Kormann, M. Grandin, and K. Virta, *Stable difference methods for block-oriented adaptive grids*, Journal of Scientific Computing, (to appear) (2014), pp. 1–26.

[14] J. Nordström, *Conservative finite difference formulations, variable coefficients, energy estimates and artificial dissipation*, Journal of Scientific Computing, 29 (2006), pp. 375–404.

[15] Jan Nordström and Mark H. Carpenter, *High-order finite difference methods, multidimensional linear problems, and curvilinear coordinates*, Journal of Computational Physics, 173 (2001), pp. 149–174.

[16] J. Nordström and J. Gong, *A stable hybrid method for hyperbolic problems*, Journal of Computational Physics, 212 (2006), pp. 436 – 453.

[17] P. Olsson, *Summation by parts, projections, and stability. II*, Mathematics of Computation, 64 (1995), pp. 1473–1493.

[18] N.A. Petersson and B. Sjögreen, *Stable grid refinement and singular source discretiztion for seismic wave simulations*, Communications in Computational Physics, 8 (2010), pp. 1074–1110.

[19] B. Strand, *Summation by parts for finite difference approximations for d/dx*, Journal of Computational Physics, 110 (1994), pp. 47–67.